
NLSY97 Appendix 3:
Family Background Variable Creation

HOUSEHOLD SIZE AS OF THE SURVEY DATE

Variables Created: CV_HH_SIZE
CV_HH_UNDER_6
CV_HH_UNDER_18

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
PUBID	PUBID	SE3001-16	SE-30.01-.16
SE1D, M, Y	SE-1_D,_M,_Y	SE3101-16	SE-31.01-.16
SE28D1-D16	SE-28.01_D-.16_D	SE3201-10	SE-32.01-.10
SE28M1-M16	SE-28.01_M-.16_M	SE3401-08	SE-34.01-.08
SE28Y1-Y16	SE-28.01_Y-.16_Y	SIDCODE	SIDCODE

This program creates several variables describing the composition of the respondent's household: the total number of residents, the number of residents under age 6, and the number of residents under age 18.

```
/* First, create a variable indicating the number of people in the household: */
array SE28D SE28D1 SE28D2 SE28D3 SE28D4 SE28D5 SE28D6 SE28D7 SE28D8 SE28D9 SE28D10
      SE28D11 SE28D12 SE28D13 SE28D14 SE28D15 SE28D16;
array SE28M SE28M1 SE28M2 SE28M3 SE28M4 SE28M5 SE28M6 SE28M7 SE28M8 SE28M9 SE28M10
      SE28M11 SE28M12 SE28M13 SE28M14 SE28M15 SE28M16;
array SE28Y SE28Y1 SE28Y2 SE28Y3 SE28Y4 SE28Y5 SE28Y6 SE28Y7 SE28Y8 SE28Y9 SE28Y10 SE28Y11
      SE28Y12 SE28Y13 SE28Y14 SE28Y15 SE28Y16;

if SE28D1 ne -4 or SE28M1 ne -4 or SE28Y1 ne -4 then dumhh1=1; else dumhh1=0;
if dumhh1=1 and SE28D2 ne -4 and SE28M2 ne -4 and SE28Y2 ne -4 then dumhh2=1; else dumhh2=0;
if dumhh2=1 and SE28D3 ne -4 and SE28M3 ne -4 and SE28Y3 ne -4 then dumhh3=1; else dumhh3=0;
if dumhh3=1 and SE28D4 ne -4 and SE28M4 ne -4 and SE28Y4 ne -4 then dumhh4=1; else dumhh4=0;
if dumhh4=1 and SE28D5 ne -4 and SE28M5 ne -4 and SE28Y5 ne -4 then dumhh5=1; else dumhh5=0;
if dumhh5=1 and SE28D6 ne -4 and SE28M6 ne -4 and SE28Y6 ne -4 then dumhh6=1; else dumhh6=0;
if dumhh6=1 and SE28D7 ne -4 and SE28M7 ne -4 and SE28Y7 ne -4 then dumhh7=1; else dumhh7=0;
if dumhh7=1 and SE28D8 ne -4 and SE28M8 ne -4 and SE28Y8 ne -4 then dumhh8=1; else dumhh8=0;
if dumhh8=1 and SE28D9 ne -4 and SE28M9 ne -4 and SE28Y9 ne -4 then dumhh9=1; else dumhh9=0;
if dumhh9=1 and SE28D10 ne -4 and SE28M10 ne -4 and SE28Y10 ne -4 then dumhh10=1; else dumhh10=0;
if dumhh10=1 and SE28D11 ne -4 and SE28M11 ne -4 and SE28Y11 ne -4 then dumhh11=1; else dumhh11=0;
if dumhh11=1 and SE28D12 ne -4 and SE28M12 ne -4 and SE28Y12 ne -4 then dumhh12=1; else dumhh12=0;
if dumhh12=1 and SE28D13 ne -4 and SE28M13 ne -4 and SE28Y13 ne -4 then dumhh13=1; else dumhh13=0;
if dumhh13=1 and SE28D14 ne -4 and SE28M14 ne -4 and SE28Y14 ne -4 then dumhh14=1; else dumhh14=0;
if dumhh14=1 and SE28D15 ne -4 and SE28M15 ne -4 and SE28Y15 ne -4 then dumhh15=1; else dumhh15=0;
if dumhh15=1 and SE28D16 ne -4 and SE28M16 ne -4 and SE28Y16 ne -4 then dumhh16=1; else dumhh16=0;

hhmember=dumhh1+dumhh2+dumhh3+dumhh4+dumhh5+dumhh6+dumhh7+dumhh8+dumhh9+dumhh10+du
mhh11+dumhh12+dumhh13+dumhh14+dumhh15+dumhh16;

/*Second, count the number of members under or above a given age. First compare their reported birth date with
the interview date and then count the number of members in the given age range. */

array age age01 age02 age03 age04 age05 age06 age07 age08 age09 age10 age11 age12 age13 age14 age15 age16;
array SE30 SE3001 SE3002 SE3003 SE3004 SE3005 SE3006 SE3007 SE3008 SE3009 SE3010 SE3011 SE3012
      SE3013 SE3014 SE3015 SE3016;
array SE31 SE3101 SE3102 SE3103 SE3104 SE3105 SE3106 SE3107 SE3108 SE3109 SE3110 SE3111 SE3112
      SE3113 SE3114 SE3115 SE3116;

do I=1 to 16;
  if SE28Y(I) eq -4 then age(I)=-4;
```

```

if SE28Y(I) eq -3 then age(I)=-3;
if SE28Y(I) eq -2 then age(I)=-2;
if SE28Y(I) eq -1 then age(I)=-1;
if SE28Y(I) gt SE1Y then age(I)=-2;
if SE28Y(I) le SE1Y and SE28Y(I) gt 0 and SE1Y gt 0 then age(I)=(SE1Y-SE28Y(I));
if age(I) ge 0 and SE28M(I) gt SE1M and SE1M gt 0 and SE28M(I) gt 0 then age(I)=(SE1Y-SE28Y(I)-1);
if age(I) ge 0 and SE28M(I) lt SE1M and SE28M(I) gt 0 and SE1M gt 0 then age(I)=(SE1Y-SE28Y(I));
if age(I) ge 0 and SE28M(I) eq SE1M and SE28D(I) gt SE1D and SE1D gt 0 and SE1D gt 0 then age(I)=(SE1Y-SE28Y(I)-1);
if age(I) ge 0 and SE28M(I) eq SE1M and SE28D(I) le SE1D and SE28D(I) gt 0 and SE1D gt 0 then
    age(I)=(SE1Y-SE28Y(I));
if SE31(I) ge 0 then age(I)=SE31(I);
end;

array dummy1 dummy101 dummy102 dummy103 dummy104 dummy105 dummy106 dummy107 dummy108
    dummy109 dummy110 dummy111 dummy112 dummy113 dummy114 dummy115 dummy116;
array dummy2 dummy201 dummy202 dummy203 dummy204 dummy205 dummy206 dummy207 dummy208
    dummy209 dummy210 dummy211 dummy212 dummy213 dummy214 dummy215 dummy216;
array dummy3 dummy301 dummy302 dummy303 dummy304 dummy305 dummy306 dummy307 dummy308
    dummy309 dummy310 dummy311 dummy312 dummy313 dummy314 dummy315 dummy316;
array SE32 SE3201 SE3202 SE3203 SE3204 SE3205 SE3206 SE3207 SE3208 SE3209 SE3210 SE3211 SE3212
    SE3213 SE3214 SE3215 SE3216;
array SE34 SE3401 SE3402 SE3403 SE3404 SE3405 SE3406 SE3407 SE3408 SE3409 SE3410 SE3411 SE3412
    SE3413 SE3414 SE3415 SE3416;

do I=1 to 16;
    if age(I) lt 6 and age(I) ge 0 then dummy1(I)=1; else dummy1(I)=0;
    if (age(I) lt 18 and age(I) ge 0) or SE32(I)=1 or SE34(I)=1 then dummy2(I)=1; else dummy2(I)=0;
end;

do I=1 to 16;
    if age(I) eq -1 then under6=-1;
    if age(I) eq -2 then under6=-2;
    if age(I) eq -3 then under6=-3;
    if age(I) eq -1 and SE32(I) ne 1 and SE34(I) ne 1 then under18=-1;
    if age(I) eq -2 and SE32(I) ne 1 and SE34(I) ne 1 then under18=-2;
    if age(I) eq -3 and SE32(I) ne 1 and SE34(I) ne 1 then under18=-3;
end;

do I=1 to 16;
under6=dummy101+dummy102+dummy103+dummy104+dummy105+dummy106+dummy107+
    dummy108+dummy109+dummy110+dummy111+dummy112+dummy113+dummy114+
    dummy115+dummy116;
under18=dummy201+dummy202+dummy203+dummy204+dummy205+dummy206+dummy207+
    dummy208+dummy209+dummy210+dummy211+dummy212+dummy213+dummy214+
    dummy215+dummy216;
end;

do I=1 to 16;
    if (SE31(I) eq -1 or SE31(I) eq -2 or SE31(I) eq -3) and (SE28M(I) eq -1 or SE28M(I) eq -2 or SE28M(I) eq -3)
        then A=1; else A=0;
end;

/* Some of the records are hand edited. */
if pubid in (1281, 1954) then do;
    hhmember=5; under6=1; under18=3; end;

```

Appendix 3: Family Background Variable Creation

```
if pubid in (1947, 2394, 3841, 3867, 3868, 4009, 4033, 4032, 4921, 4922, 6349, 6350) then do;  
    hhmember=4; under6=0; under18=2; end;  
  
if pubid in (1949, 6691) then do;  
    hhmember=2; under6=0; under18=1; end;  
  
if pubid in (1959, 8850) then do;  
    hhmember=3; under6=0; under18=1; end;  
  
if pubid=2284 then do;  
    hhmember=6; under6=1; under18=3; end;  
  
if pubid=4027 then do;  
    hhmember=4; under6=0; under18=1; end;  
  
if pubid in (4053, 6334, 6335) then do;  
    hhmember=5; under6=0; under18=3; end;  
  
if pubid=6803 or pubid=6804 then do;  
    hhmember=8; under6=1; under18=5; end;  
  
if pubid=8818 or pubid=8819 or pubid=8820 or pubid=8821 then do;  
    hhmember=6; under6=0; under18=4; end;  
  
endsas;
```

YOUTH'S MARITAL STATUS AS OF THE SURVEY DATE

Variables Created: CV_MARSTAT
CV_MARSTAT_COLLAPSED

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
M700	YMAR-700	M540011, 12	YMAR-5400.01.01, .02
M900	YMAR-900	M57001	YMAR-570001
M1200	YMAR-1200	M730011	YMAR-7300.01.01
M3100Y1, Y2	YMAR-3100.01_Y, .02_Y	M730021	YMAR-7300.02.01
M45001, 02	YMAR-4500.01, .02	PUBID	PUBID
M47001, 02	YMAR-4700.01, .02		

Codes for Created Variable

Marital/Cohabitation Status	Collapsed Marital Status
1 = never married, cohabiting	0 = never married
2 = never married, not cohabiting	1 = married
3 = married, spouse present	2 = separated
4 = married, spouse absent	3 = divorced
5 = separated, cohabiting	4 = widowed
6 = separated, not cohabiting	
7 = divorced, cohabiting	
8 = divorced, not cohabiting	
9 = widowed, cohabiting	
10 = widowed, not cohabiting	

This program creates two marital status/cohabitation status variables for respondents age 16 and older. Other respondents are valid skips (-4). Note that later partners take precedence over earlier partners.

***** COLLAPSED MARITAL STATUS VARIABLE *****.

*Initialize to not married.

COMPUTE CMARSTAT=0.

*If youth is less than 16 years old then valid skip.

IF (M700=1) CMARSTAT=-4.

*Establish whether married to partner 1. To be married R must report a partner and a valid marriage date.

IF (M1200>0 AND M45001=1 AND M3100Y1>0) MARRY1=1.

IF (M1200>0 AND M45001=0 AND (M540011=1 AND M5700Y1>0)) MARRY1=1.

*Not legally married at start (M45001=1). Relationship ended - partner 1, period 1.

IF (MARRY1=1 AND M47001=0) CMARSTAT=1.

IF (MARRY1=1 AND M47001=0 AND M540011=3 AND M5600Y11>0) CMARSTAT=2.

IF (MARRY1=1 AND M47001=0 AND M540011=4 AND M5600Y11>0) CMARSTAT=3.

IF (MARRY1=1 AND M47001=0 AND M114001=5 AND M5600Y11>0) CMARSTAT=4.

*Not legally married at start (M45001=1). Relationship ended - partner 1, period 2.

IF (MARRY1=1 AND M47001=0 AND M540012=3 AND M5600Y12>0) CMARSTAT=2.

IF (MARRY1=1 AND M47001=0 AND M540012=4 AND M5600Y12>0) CMARSTAT=3.

IF (MARRY1=1 AND M47001=0 AND M114002=5 AND M5600Y12>0) CMARSTAT=4.

*If living together continuously, no changes - partner 1, period 1.

IF (MARRY1=1 AND M47001=1 AND M730011=0) CMARSTAT=1.

*Establish whether married to second partner.

IF (M1200>0 AND M45002=1 AND M3100Y2>0) MARRY2=1.

*If living together continuously, no changes - partner 2, period 1.

IF (MARRY2=1 AND M47002=1 AND M730021=0) CMARSTAT=1.

*If can't determine beginning status (married or not).

IF (M45001=-1 OR M45002=-1 OR M3100Y1=-1 OR M3100Y2=-1) CMARSTAT=-1.

Appendix 3: Family Background Variable Creation

IF (M45001=-2 OR M45002=-2 OR M3100Y1=-2 OR M3100Y2=-2) CMARSTAT=-2.
IF (M45001=-3 OR M45002=-3 OR M3100Y1=-3 OR M3100Y2=-3) CMARSTAT=-3.
*If can't determine changes in status.
IF (CMARSTAT>0 AND (M47001=-1 OR M47002=-1 OR M540011=-1)) CMARSTAT=-1.
IF (CMARSTAT>0 AND (M540012=-1 OR M5600Y11=-1 OR M5600Y12=-1)) CMARSTAT=-1.
IF (CMARSTAT>0 AND (M5700Y1=-1 OR M730011=-1 OR M730021=-1)) CMARSTAT=-1.
IF (CMARSTAT>0 AND (M114001=-1 OR M114002=-1)) CMARSTAT=-1.
*If can't determine changes in status.
IF (CMARSTAT>0 AND (M47001=-2 OR M47002=-2 OR M540011=-2)) CMARSTAT=-2.
IF (CMARSTAT>0 AND (M540012=-2 OR M5600Y11=-2 OR M5600Y12=-2)) CMARSTAT=-2.
IF (CMARSTAT>0 AND (M5700Y1=-2 OR M730011=-2 OR M730021=-2)) CMARSTAT=-2.
IF (CMARSTAT>0 AND (M114001=-2 OR M114002=-2)) CMARSTAT=-2.
*If can't determine changes in status.
IF (CMARSTAT>0 AND (M47001=-3 OR M47002=-3 OR M540011=-3)) CMARSTAT=-3.
IF (CMARSTAT>0 AND (M540012=-3 OR M5600Y11=-3 OR M5600Y12=-3)) CMARSTAT=-3.
IF (CMARSTAT>0 AND (M5700Y1=-3 OR M730011=-3 OR M730021=-3)) CMARSTAT=-3.
IF (CMARSTAT>0 AND (M114001=-3 OR M114002=-3)) CMARSTAT=-3.
*on roster from screener but R reports -1/-2 for # partners, don't include.
IF (M900=1 AND (M1200=-1 OR M1200=-2)) CMARSTAT=0.

***** YOUTH'S MARITAL OR COHABITATION STATUS AS OF THE SURVEY DATE *****.

/Initialize to never married, partner not present.*

COMPUTE MARCOHAB=2.
IF (CMARSTAT=0 AND M47001=1) MARCOHAB=1.
IF (CMARSTAT=0 AND M47001=0) MARCOHAB=2.
IF (CMARSTAT=1 AND M47001=1) MARCOHAB=3.
IF (CMARSTAT=1 AND M47001=0) MARCOHAB=4.
IF (CMARSTAT=2 AND M47001=1) MARCOHAB=5.
IF (CMARSTAT=2 AND M47001=0) MARCOHAB=6.
IF (CMARSTAT=3 AND M47001=1) MARCOHAB=7.
IF (CMARSTAT=3 AND M47001=0) MARCOHAB=8.
IF (CMARSTAT=4 AND M47001=1) MARCOHAB=9.
IF (CMARSTAT=4 AND M47001=0) MARCOHAB=10.
IF (CMARSTAT=0 AND M47002=1) MARCOHAB=1.
IF (CMARSTAT=0 AND M47002=0) MARCOHAB=2.
IF (CMARSTAT=1 AND M47002=1) MARCOHAB=3.
IF (CMARSTAT=1 AND M47002=0) MARCOHAB=4.
IF (CMARSTAT=2 AND M47002=1) MARCOHAB=5.
IF (CMARSTAT=2 AND M47002=0) MARCOHAB=6.
IF (CMARSTAT=3 AND M47002=1) MARCOHAB=7.
IF (CMARSTAT=3 AND M47002=0) MARCOHAB=8.
IF (CMARSTAT=4 AND M47002=1) MARCOHAB=9.
IF (CMARSTAT=4 AND M47002=0) MARCOHAB=10.

Do never married, partner 1 present.
Do never married, partner 1 not present.
Do married, spouse 1 present.
Do married, spouse 1 not present.
Do separated, partner 1 present.
Do separated, partner 1 not present.
Do divorced, partner 1 present.
Do divorced, partner 1 not present.
Do widowed, partner 1 present.
Do widowed, partner 1 not present.
Do never married, partner 2 present.
Do never married, partner 2 not present.
Do married, spouse 2 present.
Do married, spouse 2 not present.
Do separated, partner 2 present.
Do separated, partner 2 not present.
Do divorced, partner 2 present.
Do divorced, partner 2 not present.
Do widowed, partner 2 present.
Do widowed, partner 2 not present.

*Do all else.

IF (CMARSTAT=-1 OR M47001=-1 OR M47002=-1) MARCOHAB=-1.
IF (CMARSTAT=-2 OR M47001=-2 OR M47002=-2) MARCOHAB=-2.
IF (CMARSTAT=-3 OR M47001=-3 OR M47001=-3) MARCOHAB=-3.
IF (CMARSTAT=-4) MARCOHAB=-4.

*On roster from screener but R reports -1/-2 for # partners, don't include.

IF (M900=1 AND (M1200=-1 OR M1200=-2)) MARCOHAB=2.

*To correct for invalid skips.

IF (M700=-3) CMARSTAT=-3 and MARCOHAB=-3.

EXECUTE.

YOUTH'S RELATIONSHIP TO HOUSEHOLD PARENT FIGURE(S)

Variables Created: CV_HH_REL_BIRTH

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
pubid	PUBID	REL13_1-16	HHI2_REL13.02-.16
saq13	YSAQ-013	REL14_1-16	HHI2_REL14.01-.16
saq14	YSAQ-014	REL15_1-16	HHI2_REL15.01-.16
saq34	YSAQ-034	REL16_1-16	HHI2_REL16.01-.16
saq35	YSAQ-035	REL2_1-16	HHI2_REL2.01-.16
yid	YOUTH_ID.01	REL3_1-16	HHI2_REL3.01-.16
YMOMID	YOUTH_MOMID.01	REL4_1-16	HHI2_REL4.01-.16
YNRMOMID	YOUTH_NRMOMID.01	REL5_1-16	HHI2_REL5.01-.16
birthd, m, y	KEY!BDATE_D, _M, _Y	REL6_1-16	HHI2_REL6.01-.16
yAGE	KEY!AGE	REL7_1-16	HHI2_REL7.01-.16
PNRMOMID	PARYOUTH_MOMID	REL8_1-16	HHI2_REL8.01-.16
ScrD, M, Y	SE-1_D, _M, _Y	REL9_1-16	HHI2_REL9.01-.16
SN701-05	SN-70.01-.05	NAGE1-23	NONHHI AGE.01-.23
SN761-3	SN-76.01-.03	NREL1-21	NONHHI_RELATION.01-.21
SN791-3	SN-79.01-.03	PI017	PINF-017
AGE1-16	HHI2_AGE.01-.16	PI018	PINF-018
DOBD1-15	HHI2_DOB.01_D-.15_D	PI020	PINF-020
DOBM1-15	HHI2_DOB.01_M-.15_M	PI025	PINF-025
DOBY1-15	HHI2_DOB.01_Y-.15_Y	PI0455	PINF-045_005
REL1_1-16	HHI2_REL1.01-.16	PI058D, M, Y	PINF-058_D, _M, _Y
REL10_1-16	HHI2_REL10.01-10.16	gender1-16	HHI2_SEX.01-.16
REL11_1-16	HHI2_REL11.01-11.16	ngend1-23	NONHHI SEX.01-.23
REL12_1-16	HHI2_REL12.01-.16		

Codes for Created Variable

- | | |
|------------------------------------|---------------------------------|
| 1 = Both biological parents | 6 = Adoptive parent(s) |
| 2 = Two parents, biological mother | 7 = Foster parent(s) |
| 3 = Two parents, biological father | 8 = No parents, grandparents |
| 4 = Biological mother only | 9 = No parents, other relatives |
| 5 = Biological father only | 10 = Anything else |

This program creates a variable identifying the youth's relationship to the primary adults in the household.

```

array rela1 (i) rel1_1-rel1_19;
array rela2 (i) rel2_1-rel2_19;
array rela3 (i) rel3_1-rel3_19;
array rela4 (i) rel4_1-rel4_19;
array rela5 (i) rel5_1-rel5_19;
array rela6 (i) rel6_1-rel6_19;
array rela7 (i) rel7_1-rel7_19;
array rela8 (i) rel8_1-rel8_19;
array rela9 (i) rel9_1-rel9_19;
array rela10 (i) rel10_1-rel10_19;
array rela11 (i) rel11_1-rel11_19;
array rela12 (i) rel12_1-rel12_19;
array rela13 (i) rel13_1-rel13_19;
array rela14 (i) rel14_1-rel14_19;
array rela15 (i) rel15_1-rel15_19;
array rela16 (i) rel16_1-rel16_19;
array rela17 (i) rel17_1-rel17_19;

array rela18 (i) rel18_1-rel18_19;
array rela19 (i) rel19_1-rel19_19;
array parid (i) parid1-parid19;

momid=0;
domid=0;
adopdad=0;
admom=0;
fostma=0;
fostda=0;
stepma=0;
stepda=0;
husb=0;
wife=0;
grand=0;
relat=0;
nonrel=0;

```

```

do i=1 to 19;
if yid=1 then do;
  if rela1>48 and rela1<57 then do;
    parid=1;
  end;
  if rela1=47 or rela1=48 then do;
    grand=1;
  end;
  if rela1=1 or rela1=2 or (rela1>12 and rela1<19)
    then do;
      relat=1;
    end;
  if rela1=74 or rela1=78 or (rela1>81 and rela1<85)
    or rela1=86 or rela1=87 then do;
    relat=1;
  end;
  if rela1=59 or rela1=60 or rela1=68 or rela1=69 or
    rela1=85 then do;
    nonrel=1;
  end;
  if rela1=-1 or rela1=-2 or rela1=-3 then do;
    invalid=-3;
  end;
end;

if yid=2 then do;
  if rela2>48 and rela2<57 then do;
    parid=1;
  end;
  if rela2=47 or rela2=48 then do;
    grand=1;
  end;
  if rela2=1 or rela2=2 then do;
    relat=1;
  end;
  if rela2=74 or rela2=78 or (rela2>81 and rela2<85)
    or rela2=86 or rela2=87 then do;
    relat=1;
  end;
  if rela2=59 or rela2=60 or rela2=68 or rela2=69 or
    rela2=85 then do;
    nonrel=1;
  end;
  if rela2=-1 or rela2=-2 or rela2=-3 then do;
    invalid=-3;
  end;
end;

if yid=3 then do;
  if rela3>48 and rela3<57 then do;
    parid=1;
  end;
  if rela3=47 or rela3=48 then do;
    grand=1;
  end;
  if rela3=1 or rela3=2 then do;
    relat=1;
  end;
  if rela3=74 or rela3=78 or (rela3>81 and rela3<85)
    or rela3=86 or rela3=87 then do;
    relat=1;
  end;
  if rela3=59 or rela3=60 or rela3=68 or rela3=69 or
    rela3=85 then do;
    nonrel=1;
  end;
  if rela3=-1 or rela3=-2 or rela3=-3 then do;
    invalid=-3;
  end;
end;

if yid=4 then do;
  if rela4>48 and rela4<57 then do;
    parid=1;
  end;
  if rela4=47 or rela4=48 then do;
    grand=1;
  end;
  if rela4=1 or rela4=2 then do;
    relat=1;
  end;
  if rela4=74 or rela4=78 or (rela4>81 and rela4<85)
    or rela4=86 or rela4=87 then do;
    relat=1;
  end;
  if rela4=59 or rela4=60 or rela4=68 or rela4=69 or
    rela4=85 then do;
    nonrel=1;
  end;
  if rela4=-1 or rela4=-2 or rela4=-3 then do;
    invalid=-3;
  end;
end;

if yid=5 then do;
  if rela5>48 and rela5<57 then do;
    parid=1;
  end;
  if rela5=47 or rela5=48 then do;
    grand=1;
  end;
  if rela5=1 or rela5=2 then do;
    relat=1;
  end;
  if rela5=74 or rela5=78 or (rela5>81 and rela5<85)
    or rela5=86 or rela5=87 then do;
    relat=1;
  end;
  if rela5=59 or rela5=60 or rela5=68 or rela5=69 or
    rela5=85 then do;
    nonrel=1;
  end;
  if rela5=-1 or rela5=-2 or rela5=-3 then do;
    invalid=-3;
  end;
end;

```

```

        invalid=-3;
    end;
end;
end;

do i=1 to 19;
if yid=1 then do;
    if parid1=1 and rel1_1=5 then admom=1;
    if parid2=1 and rel2_1=5 then admom=2;
    if parid3=1 and rel3_1=5 then admom=3;
    if parid4=1 and rel4_1=5 then admom=4;
    if parid5=1 and rel5_1=5 then admom=5;
    if parid6=1 and rel6_1=5 then admom=6;
    if parid7=1 and rel7_1=5 then admom=7;
    if parid8=1 and rel8_1=5 then admom=8;
    if parid9=1 and rel9_1=5 then admom=9;
    if parid10=1 and rel10_1=5 then admom=10;
    if parid11=1 and rel11_1=5 then admom=11;
    if parid12=1 and rel12_1=5 then admom=12;
    if parid13=1 and rel13_1=5 then admom=13;
    if parid14=1 and rel14_1=5 then admom=14;
    if parid15=1 and rel15_1=5 then admom=15;
    if parid16=1 and rel16_1=5 then admom=16;
    if parid17=1 and rel17_1=5 then admom=17;
    if parid18=1 and rel18_1=5 then admom=18;
    if parid19=1 and rel19_1=5 then admom=19;
end;

if yid=2 then do;
    if parid1=1 and rel1_2=5 then admom=1;
    /* etc. through rel19_2=5 and admom=19 */
end;
if yid=5 then do;
    if parid1=1 and rel1_5=5 then admom=1;
    /* etc. through rel19_5=5 and admom=19 */
end;
if yid=4 then do;
    if parid1=1 and rel1_4=5 then admom=1;
    /* etc. through rel19_4=5 and admom=19 */
end;
if yid=5 then do;
    if parid1=1 and rel1_5=5 then admom=1;
    /* etc. through rel19_5=5 and admom=19 */
end;
if yid=1 to 19;
if yid=1 then do;
    if parid1=1 and rel1_1=6 then adopdad=1;
    /* etc. through rel19_1=6 and adopdad=19 */
end;
if yid=2 then do;
    if parid1=1 and rel1_2=6 then adopdad=1;
    /* etc. through rel19_2=6 and adopdad=19 */
end;
if yid=3 then do;
    if parid1=1 and rel1_3=6 then adopdad=1;
    /* etc. through rel19_3=6 and adopdad=19 */
end;
if yid=4 then do;
    if parid1=1 and rel1_4=6 then adopdad=1;
    /* etc. through rel19_4=6 and adopdad=19 */
end;
if yid=5 then do;
    if parid1=1 and rel1_5=6 then adopdad=1;
    /* etc. through rel19_5=6 and adopdad=19 */
end;
if yid=1 to 19;
if yid=1 then do;
    if parid1=1 and rel1_1=3 then momid=1;
    /* etc. through rel19_1=3 and momid=19 */
end;
if yid=2 then do;
    if parid1=1 and rel1_2=3 then momid=1;
    /* etc. through rel19_2=3 and momid=19 */
end;
if yid=3 then do;
    if parid1=1 and rel1_3=3 then momid=1;
    /* etc. through rel19_3=3 and momid=19 */
end;
if yid=4 then do;
    if parid1=1 and rel1_4=3 then momid=1;
    /* etc. through rel19_4=3 and momid=19 */
end;
if yid=5 then do;
    if parid1=1 and rel1_5=3 then momid=1;
    /* etc. through rel19_5=3 and momid=19 */
end;
if yid=1 to 19;
if yid=1 then do;
    if parid1=1 and rel1_1=4 then domid=1;
    /* etc. through rel19_1=4 and domid=19 */
end;
if yid=2 then do;
    if parid1=1 and rel1_2=4 then domid=1;
    /* etc. through rel19_2=4 and domid=19 */
end;
if yid=3 then do;
    if parid1=1 and rel1_3=4 then domid=1;
    /* etc. through rel19_3=4 and domid=19 */
end;
if yid=4 then do;
    if parid1=1 and rel1_4=4 then domid=1;
    /* etc. through rel19_4=4 and domid=19 */
end;
if yid=5 then do;
    if parid1=1 and rel1_5=4 then domid=1;
    /* etc. through rel19_5=4 and domid=19 */
end;

```

```

if yid=1 then do;
  if parid1=1 and rel1_1=9 then fostma=1;
    /* etc. through rel19_1=9 and fostma=19 */
  end;
if yid=2 then do;
  if parid1=1 and rel1_2=9 then fostma=1;
    /* etc. through rel19_2=9 and fostma=19 */
  end;
if yid=3 then do;
  if parid1=1 and rel1_3=9 then fostma=1;
    /* etc. through rel19_3=9 and fostma=19 */
  end;
if yid=4 then do;
  if parid1=1 and rel1_4=9 then fostma=1;
    /* etc. through rel19_4=9 and fostma=19 */
  end;
if yid=5 then do;
  if parid1=1 and rel1_5=9 then fostma=1;
    /* etc. through rel19_5=9 and fostma=19 */
  end;
end;

do i=1 to 19;
  if yid=1 then do;
    if parid1=1 and rel1_1=10 then fostda=1;
      /* etc. through rel19_1=10 and fostda=19 */
    end;
  if yid=2 then do;
    if parid1=1 and rel1_2=10 then fostda=1;
      /* etc. through rel19_2=10 and fostda=19 */
    end;
  if yid=3 then do;
    if parid1=1 and rel1_3=10 then fostda=1;
      /* etc. through rel19_3=10 and fostda=19 */
    end;
  if yid=4 then do;
    if parid1=1 and rel1_4=10 then fostda=1;
      /* etc. through rel19_4=10 and fostda=19 */
    end;
  if yid=5 then do;
    if parid1=1 and rel1_5=10 then fostda=1;
      /* etc. through rel19_5=10 and fostda=19 */
    end;
end;

do i=1 to 19;
  if yid=1 then do;
    if parid1=1 and rel1_1=7 then stepma=1;
      /* etc. through rel19_1=7 and stepma=19 */
    end;
  if yid=2 then do;
    if parid1=1 and rel1_2=7 then stepma=1;
      /* etc. through rel19_2=7 and stepma=19 */
    end;
  if yid=3 then do;
    if parid1=1 and rel1_3=7 then stepma=1;
      /* etc. through rel19_3=7 and stepma=19 */
    end;
  if yid=4 then do;
    if parid1=1 and rel1_4=7 then stepma=1;
      /* etc. through rel19_4=7 and stepma=19 */
    end;
  if yid=5 then do;
    if parid1=1 and rel1_5=7 then stepma=1;
      /* etc. through rel19_5=7 and stepma=19 */
    end;
end;

```

Appendix 3: Family Background Variable Creation

```
end;  
  
do i=1 to 19;  
  if domid>0 then do;  
    if domid=1 and rela1=2 then wife=1;  
    if domid=2 and rela2=2 then wife=2;  
    if domid=3 and rela3=2 then wife=3;  
    if domid=4 and rela4=2 then wife=4;  
    if domid=5 and rela5=2 then wife=5;  
    if domid=6 and rela6=2 then wife=6;  
    if domid=7 and rela7=2 then wife=7;  
    if domid=8 and rela8=2 then wife=8;  
    if domid=9 and rela9=2 then wife=9;  
    if domid=10 and rela10=2 then wife=10;  
    if domid=11 and rela11=2 then wife=11;  
    if domid=12 and rela12=2 then wife=12;  
    if domid=13 and rela13=2 then wife=13;  
    if domid=14 and rela14=2 then wife=14;  
    if domid=15 and rela15=2 then wife=15;  
    if domid=16 and rela16=2 then wife=16;  
    if domid=17 and rela17=2 then wife=17;  
    if domid=18 and rela18=2 then wife=18;  
    if domid=19 and rela19=2 then wife=19;  
  end;  
end;  
  
rel=10;  
  
if invalid=-3 then do;  
  rel=-3; end;  
  
if nonrel>0 then do;  
  rel=10; end;  
  
  if relat>0 then do;  
    rel=9; end;  
  
  if grand>0 and momid=0 and domid=0 then do;  
    rel=8; end;  
  
  if fostda>0 or fostma>0 then do;  
    rel=7; end;  
  
  if admom>0 or adopdad>0 then do;  
    rel=6; end;  
  
  if domid>0 and wife=0 and momid=0 then do;  
    rel=5; end;  
  
  if momid>0 and husb=0 and domid=0 then do;  
    rel=4; end;  
  
  if domid>0 and momid=0 then do;  
    if wife>0 or admom>0 or stepma>0 then rel=3; end;  
  
  if momid>0 and domid=0 then do;  
    if husb>0 or adopdad>0 or stepda>0 then rel=2;  
    end;  
  
  if momid>0 and domid>0 then do;  
    both=1;  
    rel=1; end;  
  
*hand edits;  
if pubid in (287, 288, 289, 1470, 4127, 5345, 5346)  
  then rel=1;  
  
endsas;
```

YOUTH'S FERTILITY AND CHILD STATUS

Variables Created:	CV_CHILD_BIRTH_DATE.xx_M CV_CHILD_DEATH_DATE.xx_M CV_CHILD_BIRTH_MONTH.xx CV_CHILD_STATUS.xx CV_BIO_CHILD_HH	CV_CHILD_BIRTH_DATE.xx_Y CV_CHILD_DEATH_DATE.xx_Y CV_CHILD_BIRTH_MONTH.xx CV_BIO_CHILD_NR
---------------------------	--	--

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
F300	YFER-300	BDAYD1-D4	BIOCHILD_BDATE.01_D-.04_D
F7200M1, Y1	YFER-7200M1, Y1	BDAYM1-M4	BIOCHILD_BDATE.01_M-.04_M
F7200M2, Y2	YFER-7200M2, Y2	BDAYY1-Y4	BIOCHILD_BDATE.01_Y-.04_Y
F126001-04	YFER-12600.01-.04	BDEAD1, 2	BIOCHILD_DEAD.01, .02
F128001, 02	YFER-12800.01, .02	RESHM1-4	BIOCHILD_RESIDE.01-04
F129001-04	YFER-12900.01-.04	PUBID	PUBID

Codes for Created Variables

Date of birth and death variables

Date variables are presented as both the actual month and year and the month number in a continuous month scheme.

Status variables

- 1 Adopted
- 2 Deceased
- 3 Non-resident, foster care
- 4 Non-resident, not adopted or in foster care
- 5 Resident

This program creates a number of variables describing the youth's fertility and the current status of the youth's children. For more information on the continuous month system, see appendix 7 in this document.

/* First, create a variable indicating the DOBM(I) and DOBY(I) for each biological child. Information is taken from the fertility roster (CHIL) and the fertility section of the youth survey (YFER). */

```

array BDAYM[4] BDAYM1-BDAYM4;
array BDAYY[4] BDAYY1-BDAYY4;
array DOBM[4] DOBM1-DOBM4 ;
array DOBY[4] DOBY1-DOBY4 ;

do I=1 to 4;
DOBM(I)=-4;
if BDAYM(I) eq -3 or F300 eq -3 then DOBM(I)=-3;
if BDAYM(I) eq -1 then DOBM(I)=-1;
if BDAYM(I) eq -2 then DOBM(I)=-2;
if BDAYM(I) gt 0 then DOBM(I)=BDAYM(I);
end;

a=0;
do i=1 to 3 while(DOBM(i) eq .);
a=a+1;
end;
if a ne 0 then do;
do i=1 to 4;
if (i+a) le 4 then do;
DOBM(i)=DOBM(i+a);
DOBM(i+a)=.;
end;
end;

```

Appendix 3: Family Background Variable Creation

end;

```
do I=1 to 4;
DOBY(I)=-4;
if BDAYY(I) eq -3 or F300 eq -3 then DOBY(I)=-3;
if BDAYY(I) eq -1 then DOBY(I)=-1;
if BDAYY(I) eq -2 then DOBY(I)=-2;
if BDAYY(I) gt 0 then DOBY(I)=BDAYY(I);
end;
```

```
b=0;
do i=1 to 3 while(DOBY(i) eq .);
b=b+1;
end;
if b ne 0 then do;
do i=1 to 4;
if (i+b) le 4 then do;
DOBY(i)=DOBY(i+b);
DOBY(i+b)=.;
end;
end;
end;
```

/* Second, create a continuous month scheme variable for the month of birth of the children using the formula:
(12*(DOBY(I)-1980)+DOBM(I)) */

```
array MOB[4] MOB1-MOB4 ;
```

```
do I=1 to 4;
MOB(I)=-4;
if DOBM(I) eq -3 or DOBY(I) eq -3 or F300 eq -3 then MOB(I)=-3;
if DOBM(I) eq -1 or DOBY(I) eq -1 then MOB(I)=-1;
if DOBM(I) eq -2 or DOBY(I) eq -2 then MOB(I)=-2;
if DOBM(I) gt 0 and DOBY(I) ge 1980 then MOB(I)=12*(DOBY(I)-1980)+DOBM(I);
end;
c=0;
do i=1 to 3 while(MOB(i) eq .);
c=c+1;
end;
if c ne 0 then do;
do i=1 to 4;
if (i+c) le 4 then do;
MOB(i)=MOB(i+c);
MOB(i+c)=.;
end;
end;
end;
```

/* Third, create an actual date variable for the date of death of the youth's children. Only the DOD of children not included in the roster is available, the ones for any children listed in the roster and possibly dead is not provided. */

```
array BDEAD[2] BDEAD1-BDEAD2;
array F7200M[2] F7200M1-F7200M2;
array F7200Y[2] F7200Y1-F7200Y2;
array DODM[2] DODM1-DODM2;
```

```

array DODY[2] DODY1-DODY2;

do I=1 to 2;
    if BDEAD(I) eq 1 then do;
        if F7200M(I) eq -4 then DODM(I)=-4;
        if F7200M(I) eq -3 or F300 eq -3 then DODM(I)=-3;
        if F7200M(I) eq -1 then DODM(I)=-1;
        if F7200M(I) eq -2 then DODM(I)=-2;
        if F7200M(I) gt 0 then DODM(I)=F7200M(I);
    end;
    else DODM(I)=-4;
end;

d=0;
do i=1 to 1 while(DODM(i) eq .);
d=d+1;
end;
if d ne 0 then do;
    do i=1 to 2;
        if (i+d) le 2 then do;
            DODM(i)=DODM(i+d);
            DODM(i+d)=.;
        end;
    end;
end;

do I=1 to 2;
    if BDEAD(I) eq 1 then do;
        if F7200Y(I) eq -4 then DODY(I)=-4;
        if F7200Y(I) eq -3 or F300 eq -3 then DODY(I)=-3;
        if F7200Y(I) eq -1 then DODY(I)=-1;
        if F7200Y(I) eq -2 then DODY(I)=-2;
        if F7200Y(I) gt 0 then DODY(I)=F7200Y(I);
    end;
    else DODY(I)=-4;
end;

e=0;
do i=1 to 1 while(DODY(i) eq .);
e=e+1;
end;
if e ne 0 then do;
    do i=1 to 2;
        if (i+e) le 2 then do;
            DODY(i)=DODY(i+e);
            DODY(i+e)=.;
        end;
    end;
end;

/* Fourth, create a continuous month scheme variable forthe month of death of the children using the formula:
(12*(DODY(I)-1980)+DODM(I)) */

array MOD[2] MOD1-MOD2;

do I=1 to 2;
if DODM(I) eq -4 or DODY(I) eq -4 then MOD(I)=-4;

```

Appendix 3: Family Background Variable Creation

```
if DODM(I) eq -3 or DODY(I) eq -3 or F300 eq -3 then MOD(I)=-3;
if DODM(I) eq -1 or DODY(I) eq -1 then MOD(I)=-1;
if DODM(I) eq -2 or DODY(I) eq -2 then MOD(I)=-2;
if DODM(I) gt 0 and DODY(I) ge 1980 then MOD(I)=12*(DODY(I)-1980)+DODM(I);
end;

f=0;
do i=1 to 1 while(MOD(i) eq .);
  f=f+1;
end;
if f ne 0 then do;
  do i=1 to 2;
    if (i+e) le 2 then do;
      MOD(i)=MOD(i+f);
      MOD(i+f)=.;
    end;
  end;
end;

/* Fifth, create a variable indicating the status of youth's first (second, third, fourth) child */

/* Indicating whether they live with the respondent: */

array F12600[4] F126001-F126004;
array F12800[4] F128001-F128004;
array F12900[4] F129001-F129004;
array RESHM[4] RESHM1-RESHM4;
array STATUS[4] STATUS1-STATUS4 ;

/*Initialize the STATUS variable and determine which kids live at home */

do I=1 to 4;
  STATUS(I)=-4;
  if MOB(I) eq -4 or RESHM(I) eq -4 then STATUS(I)=-4;
  if F300 eq -3 or RESHM(I) eq -3 then STATUS(I)=-3;
  if RESHM(I) eq 1 then STATUS(I)=5;
end;

/* If they are dead */
do I=1 to 2;
  if BDEAD(I) eq 1 then STATUS(I)=2;
end;

/* If they do not reside with the parents...*/
do I=1 to 4;
  /* If they were not adopted neither in foster care */
  if RESHM(I) eq 0 and BDEAD(I) ne 1 then STATUS(I)=4;
  /* If they were adopted */
  if (F12800(I)=1 or F12900(I)=4) and RESHM(I) eq 0 and BDEAD(I) ne 1 then STATUS(I)=1;
  /* If they were in foster care */
  if F12900(I)=5 and RESHM(I) eq 0 and BDEAD(I) ne 1 then STATUS(I)=3;
end;

g=0;
do i=1 to 3 while(STATUS(i) eq .);
  g=g+1;
end;
```

Appendix 3: Family Background Variable Creation

```
if g ne 0 then do;
do i=1 to 4;
  if (i+g) le 4 then do;
    STATUS(i)=STATUS(i+g);
    STATUS(i+g)=.:;
  end;
end;
end;

/* Sixth, the number of children ever born and residing in the household (TBIORES) */

array BIORES[4] BIORES1-BIORES4;

/*Initialize the BIORES variable and create TBIORES */

do I=1 to 4;
BIORES(I)=0;
if STATUS(I) eq 5 or RESHM(I)=1 then BIORES(I)=1;
TBIORES=BIORES1+BIORES2+BIORES3+BIORES4;
if MOB1=-4 and MOB2=-4 then TBIORES=-4;
if F300 eq -3 then TBIORES=-3;
end;

/* Seventh, the number of children ever born and not residing in the household (TBIONRES) */

array BIONRES[4] BIONRES1-BIONRES4;

do I=1 to 4;
BIONRES(I)=0;
if (STATUS(I) eq 1 or STATUS(I) eq 3 or STATUS(I) eq 4 or RESHM(I)=0) and STATUS(I) ne 2 then
BIONRES(I)=1;
TBIONRES=BIONRES1+BIONRES2+BIONRES3+BIONRES4;
if MOB1=-4 and MOB2=-4 then TBIONRES=-4;
if F300 eq -3 then TBIONRES=-3;
end;

ENDSAS;
```

MARRIAGE AND COHABITATION HISTORY

Variables Created:	CV_FIRST_MARRY_MONTH CV_FIRST_MARRY_DATE_M CV_FIRST_MARRY_DATE_Y CV_MARRIAGES_TTL	CV_FIRST_COHAB_MONTH CV_FIRST_COHAB_DATE_M CV_FIRST_COHAB_DATE_Y CV_COHAB_TTL
---------------------------	--	--

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
M5700M1, Y1	YMAR-5700.01.01_M, _Y	M1400	YMAR-1400
M3100M1, Y1	YMAR-3100.01_M, _Y	M1200	YMAR-1200
M3100M2, Y2	YMAR-3100.02_M, _Y	M45001, 02	YMAR-4500.01, .02

This program creates variables which provide the dates of the youth's first marriage and/or cohabitation in both a continuous month scheme and as actual dates (for more information on the continuous month scheme, see appendix 7 in this document). Summary variables also count the total number of marriages and cohabitutions for each youth. Note that these variables are available only for youths age 16 and older as of 12/31/96. If a respondent is cohabiting and then marries it is considered both a cohabitation and a marriage. If someone refuses or doesn't know the full date of their marriage or cohabitation, then the spell is counted in the total variables and the date variables are coded -1 or -2 as applicable.

***** MARRIAGE SECTION *****

*Compute first marriage date in continuous months since 1/1/80. Initialize to not married / valid skip.

```
COMPUTE CONTMARR = -4.  
COMPUTE MARMNTH1 = -4.  
COMPUTE MARYEAR1 = -4.
```

*<If there is a valid month (> zero) and year (> zero) marriage date, and respondent was not married when began cohabiting with partner (M45001=0) in round 1, use available marriage date data from loop 1.>

```
DO IF (M5700Y1 >= 0) AND (M5700M1 >= 0) AND (M45001=0) AND (CONTMARR = -4).  
COMPUTE CONTMARR = ((M5700Y1-1980)*12) + M5700M1.  
COMPUTE MARMNTH1 = M5700M1.  
COMPUTE MARYEAR1 = M5700Y1.  
END IF.
```

*<If they were married when started to cohabit (M45001=1), use start of cohabitation dates for the marriage date on the first loop.>

```
DO IF (M45001 = 1) AND (M3100Y1 >= 0) AND (M3100M1 >= 0) AND (CONTMARR = -4).  
COMPUTE CONTMARR = ((M3100Y1-1980)*12) + M3100M1.  
COMPUTE MARMNTH1 = M3100M1.  
COMPUTE MARYEAR1 = M3100Y1.  
END IF.
```

*<If they were married the 2nd time started to cohabit (M45002=1), use 2nd loop cohab dates for marriage date.>

```
DO IF (M45002 = 1) AND (M3100Y2 >= 0) AND (M3100M2 >= 0) AND (CONTMARR = -4).  
COMPUTE CONTMARR = ((M3100Y2-1980)*12) + M3100M2.  
COMPUTE MARMNTH1 = M3100M2.  
COMPUTE MARYEAR1 = M3100Y2.  
END IF.
```

*<Code all don't know/refused responses as appropriate>

```
IF (M5700M1=-1) AND (MARMNTH1 = -4) MARMNTH1 = -1.  
IF (M5700Y1=-1) AND (MARYEAR1 = -4) MARYEAR1 = -1.
```

```

IF (M5700M1=-2) AND (MARMNTH1 = -4) MARMNTH1 = -2.
IF (M5700Y1=-2) AND (MARYEAR1 = -4) MARYEAR1 = -2.
IF (M5700M1=-3) AND (MARMNTH1 = -4) MARMNTH1 = -3.
IF (M5700Y1=-3) AND (MARYEAR1 = -4) MARYEAR1 = -3.
IF (M3100M1=-1) AND (M45001=1) AND (MARMNTH1 = -4) MARMNTH1 = -1.
IF (M3100Y1=-1) AND (M45001=1) AND (MARYEAR1 = -4) MARYEAR1 = -1.
IF (M3100M1=-2) AND (M45001=1) AND (MARMNTH1 = -4) MARMNTH1 = -2.
IF (M3100Y1=-2) AND (M45001=1) AND (MARYEAR1 = -4) MARYEAR1 = -2.
IF (M3100M1=-3) AND (M45001=1) AND (MARMNTH1 = -4) MARMNTH1 = -3.
IF (M3100Y1=-3) AND (M45001=1) AND (MARYEAR1 = -4) MARYEAR1 = -3.
IF (M3100M2=-1) AND (M45002=1) AND (MARMNTH1 = -4) MARMNTH1 = -1.
IF (M3100Y2=-1) AND (M45002=1) AND (MARYEAR1 = -4) MARYEAR1 = -1.
IF (M3100M2=-2) AND (M45002=1) AND (MARMNTH1 = -4) MARMNTH1 = -2.
IF (M3100Y2=-2) AND (M45002=1) AND (MARYEAR1 = -4) MARYEAR1 = -2.
IF (M3100M2=-3) AND (M45002=1) AND (MARMNTH1 = -4) MARMNTH1 = -3.
IF (M3100Y2=-3) AND (M45002=1) AND (MARYEAR1 = -4) MARYEAR1 = -3.

```

* Count the total number of marriages for a respondent. Initialize to zero.

```

COMPUTE TOTMARRY = 0.
COUNT TOTMARRY = MARYEAR1 (-2, -1, 1 THRU HIGHEST).
IF (M700=1) TOTMARRY=-4.
IF (M700=-3) TOTMARRY=-3.
IF (M700=-3) MARMNTH1=-3.
IF (M700=-3) MARYEAR1=-3.
IF (M700=-3) CONTMARR=-3.

```

***** COHABITATION SECTION *****

*Compute the youth's first cohabitation in months from 1/1/80. Initialize to not cohab/valid skip.

```

COMPUTE CONTCOHB = -4.
COMPUTE CHBMNTH1 = -4.
COMPUTE CHBYEAR1 = -4.
COMPUTE CHBMNTH2 = -4.
COMPUTE CHBYEAR2 = -4.

```

*<If there is a valid month and year cohabitation date in round 1 use it.>

```

DO IF (M3100Y1 > 0) AND (M3100M1>0) AND (M45001=0) AND (CONTCOHB = -4).
COMPUTE CONTCOHB = ((M3100Y1-1980)*12) + M3100M1.
COMPUTE CHBMNTH1 = M3100M1.
COMPUTE CHBYEAR1 = M3100Y1.
END IF.

```

*<If married when started cohabiting (M45001=1), use dates for next spell of cohabitation in which respondent was not married when started to cohabit (i.e., M45002=0).>

```

DO IF (M45001=1) AND (M45002=0) AND (M3100Y1 GT 0) AND (M3100M1 GT 0).
COMPUTE CONTCOHB = ((M3100Y2-1980)*12) + M3100M2.
COMPUTE CHBMNTH1 = M3100M2.
COMPUTE CHBYEAR1 = M3100Y2.
END IF.

```

*<If respondent was not married when started to cohabit in the first loop, and had a second spell of cohabitation that did not start as a marriage, calculate the dates of the second cohabitation.>

```

DO IF (M3100Y2 GT 0) AND (M3100M2 GT 0) AND (M45001=0) AND (M45002=0).
COMPUTE CHBMNTH2 = M3100M2.

```

COMPUTE CHBYEAR2 = M3100Y2.
END IF.

*<Code all don't know/refused responses as appropriate>

```
IF (M3100M1=-1) AND (M45001=0) AND (CHBMNTH1 = -4) CHBMNTH1=-1.  
IF (M3100Y1=-1) AND (M45001=0) AND (CHBYEAR1 = -4) CHBYEAR1=-1.  
IF (M3100M1=-2) AND (M45001=0) AND (CHBMNTH1 = -4) CHBMNTH1=-2.  
IF (M3100Y1=-2) AND (M45001=0) AND (CHBYEAR1 = -4) CHBYEAR1=-2.  
IF (M3100M1=-3) AND (M45001=0) AND (CHBMNTH1 = -4) CHBMNTH1=-3.  
IF (M3100Y1=-3) AND (M45001=0) AND (CHBYEAR1 = -4) CHBYEAR1=-3.  
IF (M3100M2=-1) AND (M45001=1) AND (CHBMNTH1 = -4) CHBMNTH1=-1.  
IF (M3100Y2=-1) AND (M45001=1) AND (CHBYEAR1 = -4) CHBYEAR1=-1.  
IF (M3100M2=-2) AND (M45001=1) AND (CHBMNTH1 = -4) CHBMNTH1=-2.  
IF (M3100Y2=-2) AND (M45001=1) AND (CHBYEAR1 = -4) CHBYEAR1=-2.  
IF (M3100M2=-3) AND (M45001=1) AND (CHBMNTH1 = -4) CHBMNTH1=-3.  
IF (M3100Y2=-3) AND (M45001=1) AND (CHBYEAR1 = -4) CHBYEAR1=-3.  
IF (M3100M2=-1) AND (M45001=0) AND (M45002=0) AND (CHBMNTH1 = -4) CHBMNTH1=-1.  
IF (M3100Y2=-1) AND (M45001=0) AND (M45002=0) AND (CHBYEAR1 = -4) CHBYEAR1=-1.  
IF (M3100M2=-2) AND (M45001=0) AND (M45002=0) AND (CHBMNTH1 = -4) CHBMNTH1=-2.  
IF (M3100Y2=-2) AND (M45001=0) AND (M45002=0) AND (CHBYEAR1 = -4) CHBYEAR1=-2.  
IF (M3100M2=-3) AND (M45001=0) AND (M45002=0) AND (CHBMNTH1 = -4) CHBMNTH1=-3.  
IF (M3100Y2=-3) AND (M45001=0) AND (M45002=0) AND (CHBYEAR1 = -4) CHBYEAR1=-3.
```

*Compute the total number of cohabitations for a respondent. Initialize to zero.

```
COMPUTE TOTCOHAB = 0.  
COUNT TOTCOHAB = CHBYEAR1 CHBYEAR2 (-2, -1, 1 THRU HIGHEST).  
IF (M700=1) TOTCOHAB=-4.  
IF (M700=-3) CHBMNTH1=-3.  
IF (M700=-3) CHBMNTH2=-3.  
IF (M700=-3) CHBYEAR1=-3.  
IF (M700=-3) CHBYEAR2=-3.  
IF (M700=-3) TOTCOHAB=-3.  
IF (M700=-3) CONTCOHB=-3.  
IF (M900=1) AND (M1200<0) TOTMARRY=0.  
IF (M900=1) AND (M1200<0) TOTCOHAB=0.
```

EXECUTE.

BIOLOGICAL MOTHER'S AGE AT FIRST BIRTH/YOUTH'S BIRTH

Variables Created: CV_BIO_MOM_AGE_CHILD1
CV_BIO_MOM_AGE_YOUTH

Variables Used

This program uses the same variables as the program which creates "Youth's Relationship to Household Parent Figure(s)." These variables are listed earlier in this section.

This program creates two variables which identify the age of the youth's biological mother when she gave birth to her first child and when the youth was born.

/* First establish the relationship between youth and biomom. These are people that say they live with a bio mom - need to figure out what to do when R believe bio mom in hh but not */

```
agemomfy=-3;
smomage=-3;
momage=-3;
momid=-3;

array rela1 (i) rel1_1-rel1_19;
array rela2 (i) rel2_1-rel2_19;
array rela3 (i) rel3_1-rel3_19;
array rela4 (i) rel4_1-rel4_19;
array rela5 (i) rel5_1-rel5_19;
array rela6 (i) rel6_1-rel6_19;
array rela7 (i) rel7_1-rel7_19;
array rela8 (i) rel8_1-rel8_19;
array rela9 (i) rel9_1-rel9_19;
array rela10 (i) rel10_1-rel10_19;
array rela11 (i) rel11_1-rel11_19;
array rela12 (i) rel12_1-rel12_19;
array rela13 (i) rel13_1-rel13_19;
array rela14 (i) rel14_1-rel14_19;
array rela15 (i) rel15_1-rel15_19;
array rela16 (i) rel16_1-rel16_19;
array rela17 (i) rel17_1-rel17_19;
array rela18 (i) rel18_1-rel18_19;
array rela19 (i) rel19_1-rel19_19;
array parid (i) parid1-parid19;

/* put in corrections from parent */
do i=1 to 19;
  if yid=1 then do;
    if rela1=49 or rela1=50 then do; parid=1; end;
    end;
  if yid=2 then do;
    if rela2=49 or rela2=50 then do; parid=1; end;
    end;
  if yid=3 then do;
    if rela3=49 or rela3=50 then do; parid=1; end;
    end;
  if yid=4 then do;
    if rela4=49 or rela4=50 then do; parid=1; end;
    end;
  if yid=5 then do;
    if rela5=49 or rela5=50 then do; parid=1; end;
```

Appendix 3: Family Background Variable Creation

```
end;
end;

*correct youth's age;
if yid=1 then do;
  if birthm>scrm or (birthm=scrm and birthd>scrd) then do; age1=scry-birthy-1; end;
  if (birthm<=scrm or (birthm=scrm and birthd<=scrd)) then do; age1=scry-birthy; end;
end;
if yid=2 then do;
  if birthm>scrm or (birthm=scrm and birthd>scrd) then do; age2=scry-birthy-1; end;
  if (birthm<=scrm or (birthm=scrm and birthd<=scrd)) then do; age2=scry-birthy; end;
end;
if yid=3 then do;
  if birthm>scrm or (birthm=scrm and birthd>scrd) then do; age3=scry-birthy-1; end;
  if (birthm<=scrm or (birthm=scrm and birthd<=scrd)) then do; age3=scry-birthy; end;
end;
if yid=4 then do;
  if birthm>scrm or (birthm=scrm and birthd>scrd) then do; age4=scry-birthy-1; end;
  if (birthm<=scrm or (birthm=scrm and birthd<=scrd)) then do; age4=scry-birthy; end;
end;
if yid=5 then do;
  if birthm>scrm or (birthm=scrm and birthd>scrd) then do; age5=scry-birthy-1; end;
  if (birthm<=scrm or (birthm=scrm and birthd<=scrd)) then do; age5=scry-birthy; end;
end;

do i=1 to 19;
  if yid=1 then do;
    if parid19=1 and rel19_1=3 and gender19=2 then momid=19;
    if parid18=1 and rel18_1=3 and gender18=2 then momid=18;
    if parid17=1 and rel17_1=3 and gender17=2 then momid=17;
    if parid16=1 and rel16_1=3 and gender16=2 then momid=16;
    if parid15=1 and rel15_1=3 and gender15=2 then momid=15;
    if parid14=1 and rel14_1=3 and gender14=2 then momid=14;
    if parid13=1 and rel13_1=3 and gender13=2 then momid=13;
    if parid12=1 and rel12_1=3 and gender12=2 then momid=12;
    if parid11=1 and rel11_1=3 and gender11=2 then momid=11;
    if parid10=1 and rel10_1=3 and gender10=2 then momid=10;
    if parid9=1 and rel9_1=3 and gender9=2 then momid=9;
    if parid8=1 and rel8_1=3 and gender8=2 then momid=8;
    if parid7=1 and rel7_1=3 and gender7=2 then momid=7;
    if parid6=1 and rel6_1=3 and gender6=2 then momid=6;
    if parid5=1 and rel5_1=3 and gender5=2 then momid=5;
    if parid4=1 and rel4_1=3 and gender4=2 then momid=4;
    if parid3=1 and rel3_1=3 and gender3=2 then momid=3;
    if parid2=1 and rel2_1=3 and gender2=2 then momid=2;
    if parid1=1 and rel1_1=3 and gender1=2 then momid=1;
  end;
  if yid=2 then do;
    if parid19=1 and rel19_2=3 and gender19=2 then momid=19;
    /* and so on through parid1=1, rel1_2=3, gender1=2, and momid=1 */
  end;
  if yid=3 then do;
    if parid19=1 and rel19_3=3 and gender19=2 then momid=19;
    /* and so on through parid1=1, rel1_3=3, gender1=2, and momid=1 */
  end;
  if yid=4 then do;
    if parid19=1 and rel19_4=3 and gender19=2 then momid=19;
```

```

/* and so on through parid1=1, rel142=3, gender1=2, and momid=1 */
end;
if yid=5 then do;
  if parid19=1 and rel19_5=3 and gender19=2 then momid=19;
    /* and so on through parid1=1, rel1_5=3, gender1=2, and momid=1 */
  end;
end;

do i=1 to 19;
  if momid=.. and ymomid ne . then do;
    if yid=1 then do;
      if ymomid19=1 and rel19_1=3 and gender19=2 then momid=19;
      if ymomid18=1 and rel18_1=3 and gender18=2 then momid=18;
      if ymomid17=1 and rel17_1=3 and gender17=2 then momid=17;
      if ymomid16=1 and rel16_1=3 and gender16=2 then momid=16;
      if ymomid15=1 and rel15_1=3 and gender15=2 then momid=15;
      if ymomid14=1 and rel14_1=3 and gender14=2 then momid=14;
      if ymomid13=1 and rel13_1=3 and gender13=2 then momid=13;
      if ymomid12=1 and rel12_1=3 and gender12=2 then momid=12;
      if ymomid11=1 and rel11_1=3 and gender11=2 then momid=11;
      if ymomid10=1 and rel10_1=3 and gender10=2 then momid=10;
      if ymomid9=1 and rel9_1=3 and gender9=2 then momid=9;
      if ymomid8=1 and rel8_1=3 and gender8=2 then momid=8;
      if ymomid7=1 and rel7_1=3 and gender7=2 then momid=7;
      if ymomid6=1 and rel6_1=3 and gender6=2 then momid=6;
      if ymomid5=1 and rel5_1=3 and gender5=2 then momid=5;
      if ymomid4=1 and rel4_1=3 and gender4=2 then momid=4;
      if ymomid3=1 and rel3_1=3 and gender3=2 then momid=3;
      if ymomid2=1 and rel2_1=3 and gender2=2 then momid=2;
      if ymomid1=1 and rel1_1=3 and gender1=2 then momid=1;
    end;
    if yid=2 then do;
      if ymomid19=1 and rel19_2=3 and gender19=2 then momid=19;
        /* and so on through ymomid1=1, rel1_2=3, gender1=2, and momid=1 */
    end;
  end;
  if yid=3 then do;
    if ymomid19=1 and rel19_3=3 and gender19=2 then momid=19;
    /* and so on through ymomid1=1, rel1_3=3, gender1=2, and momid=1 */
  end;
  if yid=4 then do;
    if ymomid19=1 and rel19_4=3 and gender19=2 then momid=19;
    /* and so on through ymomid1=1, rel1_4=3, gender1=2, and momid=1 */
  end;
  if yid=5 then do;
    if ymomid19=1 and rel19_5=3 and gender19=2 then momid=19;
    /* and so on through ymomid1=1, rel1_5=3, gender1=2, and momid=1 */
  end;
end;

/* Connect the age of the mom to the correct id */
/*define for nonresident mom */

if ynrmomid=9 and ngend9=2 then momage=nage9;
if ynrmomid=8 and ngend8=2 then momage=nage8;
if ynrmomid=7 and ngend7=2 then momage=nage7;
if ynrmomid=6 and ngend6=2 then momage=nage6;

```

Appendix 3: Family Background Variable Creation

```
if ynrmomid=5 and ngend5=2 then momage=nage5;
if ynrmomid=4 and ngend4=2 then momage=nage4;
if ynrmomid=3 and ngend3=2 then momage=nage3;
if ynrmomid=2 and ngend2=2 then momage=nage2;
if ynrmomid=1 and ngend1=2 then momage=nage1;

if pnrmomid=9 and ngend9=2 then momage=nage9;
if pnrmomid=8 and ngend8=2 then momage=nage8;
if pnrmomid=7 and ngend7=2 then momage=nage7;
if pnrmomid=6 and ngend6=2 then momage=nage6;
if pnrmomid=5 and ngend5=2 then momage=nage5;
if pnrmomid=4 and ngend4=2 then momage=nage4;
if pnrmomid=3 and ngend3=2 then momage=nage3;
if pnrmomid=2 and ngend2=2 then momage=nage2;
if pnrmomid=1 and ngend1=2 then momage=nage1;

/* define for mom in household */

if momid=1 and doby1 ne -3 then momage=age1;
if momid=2 and doby2 ne -3 then momage=age2;
if momid=3 and doby3 ne -3 then momage=age3;
if momid=4 and doby4 ne -3 then momage=age4;
if momid=5 and doby5 ne -3 then momage=age5;
if momid=6 and doby6 ne -3 then momage=age6;
if momid=7 and doby7 ne -3 then momage=age7;
if momid=8 and doby8 ne -3 then momage=age8;
if momid=9 and doby9 ne -3 then momage=age9;
if momid=10 and doby10 ne -3 then momage=age10;
if momid=11 and doby11 ne -3 then momage=age11;
if momid=12 and doby12 ne -3 then momage=age12;
if momid=13 and doby13 ne -3 then momage=age13;
if momid=14 and doby14 ne -3 then momage=age14;
if momid=15 and doby15 ne -3 then momage=age15;
if momid=16 and doby16 ne -3 then momage=age16;
if momid=17 and doby17 ne -3 then momage=age17;
if momid=18 and doby18 ne -3 then momage=age18;
if momid=19 and doby19 ne -3 then momage=age19;
if momid=20 and doby20 ne -3 then momage=age20;

if pi025=0 and pi058y>0 or pi058m>0 then do;
  if momid=pi020 or (momid=parentid and pi020=-4) then do;
    if pi058y>0 and scry>1996 and (pi058m>scrm or (pi058m=scrm and pi058d>scrd)) then do;
      momage=scry-pi058y-1; agefl=1; end;
    if pi058y>0 and scry>1996 and (pi058m<=scrm or (pi058m=scrm and pi058d<=scrd)) then do;
      momage=scry-pi058y; agefl=1; end;
  end;
end;

/* loop through looking for brothers and sisters (13, 14, 15, 18), if we have these, and there are no negative value
relationships, and we have a valid year of birth on everyone, then we can find the oldest's age and subtract from
mom's age)*/

array age (i) age1-age19; array sage (i) sage1-sage19;

do i=1 to 19;
  sage=-4;
  if yid=1 and momage>0 then do;
```

```

if (rela1=13 or rela1=14 or rela1=15 or rela1=18) and age ge 0 then sage=momage-age;
if rela1=20 or rela1=17 then sage=-3;
if (rela1=13 or rela1=14 or rela1=15 or rela1=18) and age<0 then sage=-4;
end;
if yid=2 and momage>0 then do;
  if (rela2=13 or rela2=14 or rela2=15 or rela2=18) and age ge 0 then sage=momage-age;
  if rela2=20 or rela2=17 then sage=-3;
  if (rela2=13 or rela2=14 or rela2=15 or rela2=18) and age<0 then sage=-4;
end;
if yid=3 and momage>0 then do;
  if (rela3=13 or rela3=14 or rela3=15 or rela3=18) and age ge 0 then sage=momage-age;
  if rela3=20 or rela3=17 then sage=-3;
  if (rela3=13 or rela3=14 or rela3=15 or rela3=18) and age<0 then sage=-4;
end;
if yid=4 and momage>0 then do;
  if (rela4=13 or rela4=14 or rela4=15 or rela4=18) and age ge 0 then sage=momage-age;
  if rela4=20 or rela4=17 then sage=-3;
  if (rela4=13 or rela4=14 or rela4=15 or rela4=18) and age<0 then sage=-4;
end;
if yid=5 and momage>0 then do;
  if (rela5=13 or rela5=14 or rela5=15 or rela5=18) and age ge 0 then sage=momage-age;
  if rela5=20 or rela5=17 then sage=-3;
  if (rela5=13 or rela5=14 or rela5=15 or rela5=18) and age<0 then sage=-4;
end;
end;

array nrel (i) nrel1-nrel23; array nage (i) nage1-nage23; array nrsage (i) nrsage1-nrsage23;

do i=1 to 23;
  nrsage=-4;
  if momage>0 then do;
    if (nrel=13 or nrel=14 or nrel=15 or nrel=18) and nage ge 0 then nrsage=momage-nage;
    if nrel=20 or nrel=17 then nrsage=-3;
    if (nrel=13 or nrel=14 or nrel=15 or nrel=18) and nage<0 then nrsage=-3;
  end;
end;

/* To get youths birth: if mom died, find yeardif by subtracting the year of death from the Rs birthyear, then subtract
this difference from the age at the time of death to get to the bio mom's age at Rs birth */

if yid=1 and momage>0 and age1>0 then do; sage1=momage-age1; end;
if yid=2 and momage>0 and age2>0 then do; sage2=momage-age2; end;
if yid=3 and momage>0 and age3>0 then do; sage3=momage-age3; end;
if yid=4 and momage>0 and age4>0 then do; sage4=momage-age4; end;
if yid=5 and momage>0 and age5>0 then do; sage5=momage-age5; end;

unadj1=sage1;
unadj2=sage2;
unadj3=sage3;
unadj4=sage4;
unadj5=sage5;

array sn76 (i) sn761-sn763; array sn79 (i) sn791-sn793;

do i=1 to 3;
  if sn701=0 and yid=1 and (sn76 ge birthy) then do;
    yeardif=sn76-birthy; sage1=sn791-yeardif; end;

```

```

if sn702=0 and yid=2 and (sn76 ge birthy) then do;
  yeardif=sn76-birthy; sage2=sn79-yeardif; end;
if sn703=0 and yid=3 and (sn76 ge birthy) then do;
  yeardif=sn76-birthy; sage3=sn79-yeardif; end;
if sn704=0 and yid=4 and (sn76 ge birthy) then do;
  yeardif=sn76-birthy; sage4=sn79-yeardif; end;
if sn705=0 and yid=5 and (sn76 ge birthy) then do;
  yeardif=sn76-birthy; sage5=sn79-yeardif; end;
end;

array sdif (i) sdif1-sdif23;

do i=1 to 23;
  if sn701=0 and yid=1 and yeardif>0 and yeardif ne . and nrsage>0 then do;
    sdif=unadj1-nrsage; nrsage=sage1-sdif; end;
  if sn702=0 and yid=2 and yeardif>0 and yeardif ne . and nrsage>0 then do;
    sdif=unadj2-nrsage; nrsage=sage2-sdif; end;
  if sn703=0 and yid=3 and yeardif>0 and yeardif ne . and nrsage>0 then do;
    sdif=unadj3-nrsage; nrsage=sage3-sdif; end;
  if sn704=0 and yid=4 and yeardif>0 and yeardif ne . and nrsage>0 then do;
    sdif=unadj4-nrsage; nrsage=sage4-sdif; end;
  if sn705=0 and yid=5 and yeardif>0 and yeardif ne . and nrsage>0 then do;
    sdif=unadj5-nrsage; nrsage=sage5-sdif; end;
end;

array sibdif (i) sibdif1-sibdif23;
do i=1 to 19;
  sibdif=0;
end;

do i=2 to 19;
  if sn701=0 and yid=1 and yeardif>0 and yeardif ne . and sage>0 then do;
    sibdif=unadj1-sage; sage=sage1-sibdif; end;
end;
do i=3 to 19;
  if sn702=0 and yid=2 and yeardif>0 and yeardif ne . and sage>0 then do;
    sibdif=unadj2-sage; sibdif1=unadj2-unadj1; sage=sage2-sibdif; sage1=sage2-sibdif1; end;
end;
do i=4 to 19;
  if sn703=0 and yid=3 and yeardif>0 and yeardif ne . and sage>0 then do;
    sibdif=unadj3-sage; sibdif1=unadj3-unadj1; sibdif2=unadj3-unadj2;
    sage=sage3-sibdif; sage1=sage3-sibdif1; sage2=sage3-sibdif2;
  end;
end;
do i=5 to 19;
  if sn704=0 and yid=4 and yeardif>0 and yeardif ne . and sage>0 then do;
    sibdif=unadj4-sage; sibdif1=unadj4-unadj1; sibdif2=unadj4-unadj2; sibdif3=unadj4-unadj3;
    sage=sage4-sibdif; sage1=sage4-sibdif1; sage2=sage4-sibdif2; sage3=sage4-sibdif3;
  end;
end;
do i=6 to 19;
  if sn705=0 and yid=5 and yeardif>0 and yeardif ne . and sage>0 then do;
    sibdif=unadj5-sage; sibdif1=unadj5-unadj1; sibdif2=unadj5-unadj2; sibdif3=unadj5-unadj3;
    sibdif4=unadj5-unadj4;
    sage=sage5-sibdif; sage1=sage5-sibdif1; sage2=sage5-sibdif2; sage3=sage5-sibdif3; sage4=sage5-sibdif4;
  end;
end;

```

```

/* Recalculate Youth variable and age */
if yid=1 and sage1>0 then do; smomage=sage1; end;
if yid=2 and sage2>0 then do; smomage=sage2; end;
if yid=3 and sage3>0 then do; smomage=sage3; end;
if yid=4 and sage4>0 then do; smomage=sage4; end;
if yid=5 and sage5>0 then do; smomage=sage5; end;

do i=1 to 19;
    if sage=-4 then do; sage=999; end;
    if (sage<0 and sage ne -4) or sage=0 then do; sage=0; end;
end;
do i=1 to 23;
    if nrsage=-4 then do; nrsage=999; end;
    if (nrsage<0 and nrsage ne -4) or nrsage=0 then do; nrsage=0; end;
end;

if sage1>0 and sage2>0 and sage3>0 and sage4>0 and sage5>0 and sage6>0 and sage7>0 and sage8>0 and sage9>0
    and sage10>0 and sage11>0 and sage12>0 and sage13>0 and sage14>0 and sage15>0 and sage16>0 and
    sage17>0 and sage18>0 and sage19>0 and nrsage1>0 and nrsage2>0 and nrsage3>0 and nrsage4>0 and
    nrsage5>0 and nrsage6>0 and nrsage7>0 and nrsage8>0 and nrsage9>0 and nrsage10>0 and nrsage11>0
    and nrsage12>0 and nrsage13>0 and nrsage14>0 and nrsage15>0 and nrsage16>0 and nrsage17>0 and
    nrsage18>0 and nrsage19>0 and nrsage20>0 and nrsage21>0 and nrsage22>0 and nrsage23>0 then do;
    agemomfy=min(sage1, sage2, sage3, sage4, sage5, sage6, sage7, sage8, sage9, sage10, sage11, sage12, sage13,
        sage14, sage15, sage16, sage17, sage18, sage19, nrsage1, nrsage2, nrsage3, nrsage4, nrsage5, nrsage6,
        nrsage7, nrsage8, nrsage9, nrsage10, nrsage11, nrsage12, nrsage13, nrsage14, nrsage15, nrsage16, nrsage17,
        nrsage18, nrsage19, nrsage20, nrsage21, nrsage22, nrsage23);
end;

if agemomfy=999 then agemomfy=-3;

/*youth says doesn't live with bio mom and no survey information collected on non-res bio mom*/
if sn701=1 and yid=1 then do;
    if (saq13=1 and saq14 ne 1) or (saq13<1 and saq13>-4) then do;
        smomage=-3; agemomfy=-3; end;
    end;
if sn702=1 and yid=2 then do;
    if (saq13=1 and saq14 ne 1) or (saq13<1 and saq13>-4) then do;
        smomage=-3; agemomfy=-3; end;
    end;
if sn703=1 and yid=3 then do;
    if (saq13=1 and saq14 ne 1) or (saq13<1 and saq13>-4) then do;
        smomage=-3; agemomfy=-3; end;
    end;
if sn704=1 and yid=4 then do;
    if (saq13=1 and saq14 ne 1) or (saq13<1 and saq13>-4) then do;
        smomage=-3; agemomfy=-3; end;
    end;
if sn705=5 and yid=5 then do;
    if (saq13=1 and saq14 ne 1) or (saq13<1 and saq13>-4) then do;
        smomage=-3; agemomfy=-3; end;
endsas;

```

YOUTH CITIZENSHIP STATUS

Variables Created: CV_CITIZENSHIP

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
pubid	PUBID	dob01y	KEY!BDATE_Y
pinf015y	PINF-015_Y	spparid1-6	HHI2_SPOPARID.01-.16
pinf020	PINF-020	hh1rel1-5	HHI2_REL1.01-.05
pinf096	PINF-096	hh2rel1-5	HHI2_REL2.01-.05
pinf097	PINF-097	hh3rel1-5	HHI2_REL3.01-.05
pinf159	PINF-159	hh4rel1-5	HHI2_REL4.01-.05
pinf160	PINF-160	hh5rel1-5	HHI2_REL5.01-.05
y01hhid	PARYOUTH_HHID	hh6rel1-5	HHI2_REL6.01-.05
ythpar01	PARYOUTH_PARENT	hh7rel1-5	HHI2_REL7.01-.05
ythparid	PARYOUTH_PARENTID	hh8rel1-5	HHI2_REL8.01-.05
ythprsex	PARYOUTH_PARENTSEX	hh9rel1-5	HHI2_REL9.01-.05
p2001	P2-001	hh10rel1-5	HHI2_REL10.01-.05
p2002	P2-002	hh11rel1-5	HHI2_REL11.01-.05
p2008	P2-008	hh12rel1-5	HHI2_REL12.01-.05
p2037	P2-037	hh13rel1-5	HHI2_REL13.01-.05
p2038	P2-038	hh14rel1-5	HHI2_REL14.01-.05
p2044	P2-044	hh15rel1-5	HHI2_REL15.01-.05
p2108b01	P2-108B.01	hh16rel1-5	HHI2_REL16.01-.05

Codes for Created Variable

1 = Citizen, born in the U.S.

2 = Unknown, not born in the U.S.

3 = Unknown, can't determine birthplace

This program uses data from the parent interview to determine the youth's U.S. citizenship status. If the responding parent is the youth's biological parent, the responding parent's citizenship status is used to ascertain the youth's citizenship status. If the responding parent is not a bio parent, the nonresponding bio parent's citizenship status is used to determine the youth's status when available. If the responding parent is not a bio parent and nonresponding bio parent information is not available, the youth's citizenship is coded as "can't determine."

**** PARENT IDENTIFICATION SECTION ****

*<Initialize everyone to zero>

Compute parelate=0.

*<ythparid=1>

```
do if (ythparid=1 and pinf020<0) or (pinf020=1).
if (y01hhid=2) and (spparid1=3) parelate=hh3rel2.
if (y01hhid=2) and (spparid1=4) parelate=hh4rel2.
if (y01hhid=2) and (spparid1=5) parelate=hh5rel2.
if (y01hhid=2) and (spparid1=6) parelate=hh6rel2.
if (y01hhid=2) and (spparid1=7) parelate=hh7rel2.
if (y01hhid=2) and (spparid1=8) parelate=hh8rel2.
if (y01hhid=2) and (spparid1=9) parelate=hh9rel2.
if (y01hhid=2) and (spparid1=10) parelate=hh10rel2.
if (y01hhid=2) and (spparid1=11) parelate=hh11rel2.
if (y01hhid=2) and (spparid1=12) parelate=hh12rel2.
if (y01hhid=2) and (spparid1=13) parelate=hh13rel2.
if (y01hhid=2) and (spparid1=14) parelate=hh14rel2.
if (y01hhid=2) and (spparid1=15) parelate=hh15rel2.
```

```
if (y01hhid=2) and (spparid1=16) parelate=hh16rel2.
if (y01hhid=3) and (spparid1=2) parelate=hh2rel3.
if (y01hhid=3) and (spparid1=4) parelate=hh4rel3.
if (y01hhid=3) and (spparid1=5) parelate=hh5rel3.
if (y01hhid=3) and (spparid1=6) parelate=hh6rel3.
if (y01hhid=3) and (spparid1=7) parelate=hh7rel3.
if (y01hhid=3) and (spparid1=8) parelate=hh8rel3.
if (y01hhid=3) and (spparid1=9) parelate=hh9rel3.
if (y01hhid=3) and (spparid1=10) parelate=hh10rel3.
if (y01hhid=3) and (spparid1=11) parelate=hh11rel3.
if (y01hhid=3) and (spparid1=12) parelate=hh12rel3.
if (y01hhid=3) and (spparid1=13) parelate=hh13rel3.
if (y01hhid=3) and (spparid1=14) parelate=hh14rel3.
if (y01hhid=3) and (spparid1=15) parelate=hh15rel3.
if (y01hhid=3) and (spparid1=16) parelate=hh16rel3.
if (y01hhid=4) and (spparid1=2) parelate=hh2rel4.
if (y01hhid=4) and (spparid1=3) parelate=hh3rel4.
if (y01hhid=4) and (spparid1=5) parelate=hh5rel4.
if (y01hhid=4) and (spparid1=6) parelate=hh6rel4.
if (y01hhid=4) and (spparid1=7) parelate=hh7rel4.
```

```

if (y01hhid=4) and (spparid1=8) parelate=hh8rel4.
if (y01hhid=4) and (spparid1=9) parelate=hh9rel4.
if (y01hhid=4) and (spparid1=10) parelate=hh10rel4.
if (y01hhid=4) and (spparid1=11) parelate=hh11rel4.
if (y01hhid=4) and (spparid1=12) parelate=hh12rel4.
if (y01hhid=4) and (spparid1=13) parelate=hh13rel4.
if (y01hhid=4) and (spparid1=14) parelate=hh14rel4.
if (y01hhid=4) and (spparid1=15) parelate=hh15rel4.
if (y01hhid=4) and (spparid1=16) parelate=hh16rel4.
if (y01hhid=5) and (spparid1=2) parelate=hh2rel5.
if (y01hhid=5) and (spparid1=3) parelate=hh3rel5.
if (y01hhid=5) and (spparid1=4) parelate=hh4rel5.
if (y01hhid=5) and (spparid1=6) parelate=hh6rel5.
if (y01hhid=5) and (spparid1=7) parelate=hh7rel5.
if (y01hhid=5) and (spparid1=8) parelate=hh8rel5.
if (y01hhid=5) and (spparid1=9) parelate=hh9rel5.
if (y01hhid=5) and (spparid1=10) parelate=hh10rel5.
if (y01hhid=5) and (spparid1=11) parelate=hh11rel5.
if (y01hhid=5) and (spparid1=12) parelate=hh12rel5.
if (y01hhid=5) and (spparid1=13) parelate=hh13rel5.
if (y01hhid=5) and (spparid1=14) parelate=hh14rel5.
if (y01hhid=5) and (spparid1=15) parelate=hh15rel5.
if (y01hhid=5) and (spparid1=16) parelate=hh16rel5.

```

*< Due to space restrictions, the remaining relationship loops are not reproduced in their entirety. The pattern for each loop is similar to that seen above in ythparid=1. Users who need the complete code for this variable should contact NLS User Services.>

```

*<ythparid=2>
else if (ythparid=2 and pinf020<0) or (pinf020=2).
if (y01hhid=1) and (spparid2=3) parelate=hh3rel1.
    *< and so on through spparid2=16 and hh16rel1 >
if (y01hhid=3) and (spparid2=1) parelate=hh1rel3.
    *< and so on through spparid2=16 and hh16rel3 >
if (y01hhid=4) and (spparid2=1) parelate=hh1rel4.
    *< and so on through spparid2=16 and hh16rel4 >
if (y01hhid=5) and (spparid2=1) parelate=hh1rel5.
    *< and so on through hh16rel5 >

```

```

*<ythparid=3>
else if (ythparid=3 and pinf020<0) or (pinf020=3).
if (y01hhid=1) and (spparid3=2) parelate=hh2rel1.
    *< and so on through spparid3=16 and hh16rel1 >
if (y01hhid=2) and (spparid3=1) parelate=hh1rel2.
    *< and so on through spparid3=16 and hh16rel2 >
if (y01hhid=4) and (spparid3=1) parelate=hh1rel4.
    *< and so on through spparid3=16 and hh16rel4 >
if (y01hhid=5) and (spparid3=1) parelate=hh1rel5.
    *< and so on through spparid3=16 and hh16rel5 >

```

```

*<ythparid=4>
else if (ythparid=4 and pinf020<0) or (pinf020=4).
if (y01hhid=1) and (spparid4=2) parelate=hh2rel1.
    *< and so on through spparid4=16 and hh16rel1 >
if (y01hhid=2) and (spparid4=1) parelate=hh1rel2.

```

```

    *< and so on through spparid4=16 and hh16rel2 >
if (y01hhid=3) and (spparid4=1) parelate=hh1rel3.
    *< and so on through spparid4=16 and hh16rel3 >
if (y01hhid=5) and (spparid4=1) parelate=hh1rel5.
    *< and so on through spparid4=16 and hh16rel5 >

```

```

*<ythparid=5>
else if ((ythparid=5 and pinf020<0) or (pinf020=5)).
if (y01hhid=1) and (spparid5=2) parelate=hh2rel1.
    *< and so on through spparid5=16 and hh16rel1 >
if (y01hhid=2) and (spparid5=1) parelate=hh1rel2.
    *< and so on through spparid5=16 and hh16rel2 >
if (y01hhid=3) and (spparid5=1) parelate=hh1rel3.
    *< and so on through spparid5=16 and hh16rel3 >
if (y01hhid=4) and (spparid5=1) parelate=hh1rel4.
    *< and so on through spparid5=16 and hh16rel4 >

```

```

*<ythparid=6>
else if ((ythparid=6 and pinf020<0) or (pinf020=6)).
if (y01hhid=1) and (spparid6=2) parelate=hh2rel1.
    *< and so on through spparid6=16 and hh16rel1 >
if (y01hhid=2) and (spparid6=1) parelate=hh1rel2.
    *< and so on through spparid6=16 and hh16rel2 >
if (y01hhid=3) and (spparid6=1) parelate=hh1rel3.
    *< and so on through spparid6=16 and hh16rel3 >
if (y01hhid=4) and (spparid6=1) parelate=hh1rel4.
    *< and so on through spparid6=16 and hh16rel4 >
if (y01hhid=5) and (spparid6=1) parelate=hh1rel5.
    *< and so on through spparid6=16 and hh16rel5 >

```

```

*<ythparid=7>
else if ((ythparid=7 and pinf020<0) or (pinf020=7)).
if (y01hhid=1) and (spparid7=2) parelate=hh2rel1.
    *< and so on through spparid7=16 and hh16rel1 >
if (y01hhid=2) and (spparid7=1) parelate=hh1rel2.
    *< and so on through spparid7=16 and hh16rel2 >
if (y01hhid=3) and (spparid7=1) parelate=hh1rel3.
    *< and so on through spparid7=16 and hh16rel3 >
if (y01hhid=4) and (spparid7=1) parelate=hh1rel4.
    *< and so on through spparid7=16 and hh16rel4 >
if (y01hhid=5) and (spparid7=1) parelate=hh1rel5.
    *< and so on through spparid7=16 and hh16rel5 >

```

```

*<ythparid=8>
else if ((ythparid=8 and pinf020<0) or (pinf020=8)).
if (y01hhid=1) and (spparid8=2) parelate=hh2rel1.
    *< and so on through spparid8=16 and hh16rel1 >
if (y01hhid=2) and (spparid8=1) parelate=hh1rel2.
    *< and so on through spparid8=16 and hh16rel2 >
    *< and so on through spparid8=16 and hh16rel3 >
if (y01hhid=4) and (spparid8=1) parelate=hh1rel4.
    *< and so on through spparid8=16 and hh16rel4 >
if (y01hhid=5) and (spparid8=1) parelate=hh1rel5.
    *< and so on through spparid8=16 and hh16rel5 >

```

```

*<ythparid=9>
else if ((ythparid=9 and pinf020<0) or (pinf020=9)).

```

```

if (y01hhid=1) and (spparid9=2) parelate=hh2rel1.
  *< and so on through spparid9=16 and hh16rel1 >
if (y01hhid=2) and (spparid9=1) parelate=hh1rel2.
  *< and so on through spparid9=16 and hh16rel2 >
if (y01hhid=3) and (spparid9=1) parelate=hh1rel3.
  *< and so on through spparid9=16 and hh16rel3 >
if (y01hhid=4) and (spparid9=1) parelate=hh1rel4.
  *< and so on through spparid9=16 and hh16rel4 >
if (y01hhid=5) and (spparid9=1) parelate=hh1rel5.
  *< and so on through spparid9=16 and hh16rel5 >

```

******* PARENT STATUS SECTION *******

*< Determine legal status of partner/spouse of responding parent>

*< initialize everyone to valid skip>
compute slegmom = -4.
compute slegdad = -4.

*< use data from parelate to determine legal status of spouse/partner of rp>
if (parelate=-1 and ythprsex=2) slegdad=-1.
if (parelate=-2 and ythprsex=2) slegdad=-2.
if (parelate=-1 and ythprsex=1) slegmom=-1.
if (parelate=-2 and ythprsex=1) slegmom=-2.
if (parelate=3) slegdad=1.
if (parelate=4) slegmom=1.
if (parelate=5) slegdad=3.
if (parelate=6) slegmom=3.
if (parelate=7) slegdad=2.
if (parelate=8) slegmom=2.
if (parelate=9) slegdad=4.
if (parelate=10) slegmom=4.
if (parelate>10 and ythprsex=1) slegdad=4.
if (parelate>10 and ythprsex=2) slegmom=4.

*< Determine legal status of responding parent>

*<initialize everyone to valid skip>
compute legmom = -4.
compute legdad = -4.

*<include loops to account for corrections to the responding parent's hhid>

*<updated parent id = 1>
do if (pinf020=1) and (y01hhid=2).
 if (hh1rel2=3) legmom=1.
 if (hh1rel2=4) legdad=1.
 if (hh1rel2=5) legmom=3.
 if (hh1rel2=6) legdad=3.
 if (hh1rel2=7) legmom=2.
 if (hh1rel2=8) legdad=2.
 if (hh1rel2=9) legmom=4.
 if (hh1rel2=10) legdad=4.
 if (hh1rel2>10) legmom=13.

```

else if (pinf020=1) and (y01hhid=3).
  if (hh1rel3=3) legmom=1.
  if (hh1rel3=4) legdad=1.
  if (hh1rel3=5) legmom=3.
  if (hh1rel3=6) legdad=3.
  if (hh1rel3=7) legmom=2.
  if (hh1rel3=8) legdad=2.
  if (hh1rel3=9) legmom=4.
  if (hh1rel3=10) legdad=4.
  if (hh1rel3>10 and ythprsex=2) legmom=4.
  if (hh1rel3>10 and ythprsex=1) legdad=4.
else if (pinf020=1) and (y01hhid=4).
  if (hh1rel4=3) legmom=1.
  if (hh1rel4=4) legdad=1.
  if (hh1rel4=5) legmom=3.
  if (hh1rel4=6) legdad=3.
  if (hh1rel4=7) legmom=2.
  if (hh1rel4=8) legdad=2.
  if (hh1rel4=9) legmom=4.
  if (hh1rel4=10) legdad=4.
  if (hh1rel4>10 and ythprsex=2) legmom=4.
  if (hh1rel4>10 and ythprsex=1) legdad=4.
else if (pinf020=1) and (y01hhid=5).
  if (hh1rel5=3) legmom=1.
  if (hh1rel5=4) legdad=1.
  if (hh1rel5=5) legmom=3.
  if (hh1rel5=6) legdad=3.
  if (hh1rel5=7) legmom=2.
  if (hh1rel5=8) legdad=2.
  if (hh1rel5=9) legmom=4.
  if (hh1rel5=10) legdad=4.
  if (hh1rel5>10 and ythprsex=2) legmom=4.
  if (hh1rel5>10 and ythprsex=1) legdad=4.

```

*< At this point the program loops through the same 10 statements for each combination of youth and parent id. These loops are truncated here due to space considerations. Users who need the entire program should contact NLS User Services. >

*<updated parent id = 2>
else if (pinf020=2) and (y01hhid=1).
 if (hh2rel1=3) legmom=1.
else if (pinf020=2) and (y01hhid=3).
 if (hh2rel3=3) legmom=1. *< and so on>
else if (pinf020=2) and (y01hhid=4).
 if (hh2rel4=3) legmom=1. *< and so on>
else if (pinf020=2) and (y01hhid=5).
 if (hh2rel5=3) legmom=1. *< and so on>

*<updated parent id = 3>
else if (pinf020=3) and (y01hhid=1).
 if (hh3rel1=3) legmom=1. *< and so on>
else if (pinf020=3) and (y01hhid=2).
 if (hh3rel2=3) legmom=1. *< and so on>
else if (pinf020=3) and (y01hhid=4).
 if (hh3rel4=3) legmom=1. *< and so on>

```

else if (pinf020=3) and (y01hhid=5).
    if (hh3rel5=3) legmom=1. *< and so on>

*<updated parent id = 4>
else if (pinf020=4) and (y01hhid=1).
    if (hh4rel1=3) legmom=1. *< and so on>
else if (pinf020=4) and (y01hhid=2).
    if (hh4rel2=3) legmom=1. *< and so on>
else if (pinf020=4) and (y01hhid=3).
    if (hh4rel3=3) legmom=1. *< and so on>
else if (pinf020=4) and (y01hhid=5).
    if (hh4rel5=3) legmom=1. *< and so on>

*<updated parent id = 5*>< and so on>
else if (pinf020=5) and (y01hhid=1).
    if (hh5rel1=3) legmom=1. *< and so on>
else if (pinf020=5) and (y01hhid=2).
    if (hh5rel2=3) legmom=1. *< and so on>
else if (pinf020=5) and (y01hhid=3).
    if (hh5rel3=3) legmom=1. *< and so on>
else if (pinf020=5) and (y01hhid=4).
    if (hh5rel4=3) legmom=1. *< and so on>

*<updated parent id = 6>
else if (pinf020=6) and (y01hhid=1).
    if (hh6rel1=3) legmom=1. *< and so on>
else if (pinf020=6) and (y01hhid=2).
    if (hh6rel2=3) legmom=1. *< and so on>
else if (pinf020=6) and (y01hhid=3).
    if (hh6rel3=3) legmom=1. *< and so on>
else if (pinf020=6) and (y01hhid=4).
    if (hh6rel4=3) legmom=1. *< and so on>
else if (pinf020=6) and (y01hhid=5).
    if (hh6rel5=3) legmom=1. *< and so on>

*<updated parent id = 7>
else if (pinf020=7) and (y01hhid=1).
    if (hh7rel1=3) legmom=1. *< and so on>
else if (pinf020=7) and (y01hhid=2).
    if (hh7rel2=3) legmom=1. *< and so on>
else if (pinf020=7) and (y01hhid=3).
    if (hh7rel3=3) legmom=1. *< and so on>
else if (pinf020=7) and (y01hhid=4).
    if (hh7rel4=3) legmom=1. *< and so on>
else if (pinf020=7) and (y01hhid=5).
    if (hh7rel5=3) legmom=1. *< and so on>

*<updated parent id = 8>
else if (pinf020=8) and (y01hhid=1).
    if (hh8rel1=3) legmom=1. *< and so on>
else if (pinf020=8) and (y01hhid=2).
    if (hh8rel2=3) legmom=1. *< and so on>
else if (pinf020=8) and (y01hhid=3).
    if (hh8rel3=3) legmom=1. *< and so on>
else if (pinf020=8) and (y01hhid=4).
    if (hh8rel4=3) legmom=1. *< and so on>
else if (pinf020=8) and (y01hhid=5).

if (hh8rel5=3) legmom=1. *< and so on>

*<updated parent id = 9>
else if (pinf020=9) and (y01hhid=1).
    if (hh9rel1=3) legmom=1. *< and so on>
else if (pinf020=9) and (y01hhid=2).
    if (hh9rel2=3) legmom=1. *< and so on>
else if (pinf020=9) and (y01hhid=3).
    if (hh9rel3=3) legmom=1. *< and so on>
else if (pinf020=9) and (y01hhid=4).
    if (hh9rel4=3) legmom=1. *< and so on>
else if (pinf020=9) and (y01hhid=5).
    if (hh9rel5=3) legmom=1. *< and so on>

*<if there is valid data for the mother, use it.>
if (ythpar01=-1 and pinf020=-4 and pinf097=-4 and
    pinf160=-4) and (ythprsex=2) legmom=-1.
if (ythpar01=-2 and pinf020=-4 and pinf097=-4 and
    pinf160=-4) and (ythprsex=2) legmom=-2.
do if any(ythpar01, 0,1,3,5,7,9,11,13,15,17,19,21) and
    (pinf097=-4 and pinf160=-4) and (legmom=-4) and
    (ythprsex=2).
    if (ythpar01=1) legmom=1.
    if (ythpar01=3) legmom=2.
    if (ythpar01=5) legmom=3.
    if (ythpar01>=7) legmom=4.
else if (pinf097>0 or pinf160>0).
    if (pinf097=1 or pinf160=1) legmom=1.
    if (pinf097=3 or pinf160=3) legmom=2.
    if (pinf097=5 or pinf160=5) legmom=3.
    if (pinf097>=7 or pinf160>=7) legmom=4.
end if.

*<use info from youth roster if paryouth info not
available>
do if any(paryth01, 0,1,3,5,7,9,11,13,15,17,19,21) and
    (pinf097=-4 and pinf160=-4) and (legmom=-4) and
    (ythpar01=-4) and (ythparid=-4).
    if (paryth01=1) legmom=1.
    if (paryth01=3) legmom=2.
    if (paryth01=5) legmom=3.
    if (paryth01>=7) legmom=4.
end if.

*<if there is valid data for the father, use it.>
if (ythpar01=-1 and pinf020=-4 and pinf096=-4 and
    pinf159=-4) and (ythprsex=1) legdad=-1.
if (ythpar01=-2 and pinf020=-4 and pinf096=-4 and
    pinf159=-4) and (ythprsex=1) legdad=-2.
do if any(ythpar01, 0,2,4,6,8,10,12,14,16,18,20,22)
    and (pinf096=-4 and pinf159=-4) and (legdad=-4) and
    (ythprsex=1).
    if (ythpar01=2) legdad=1.
    if (ythpar01=4) legdad=2.
    if (ythpar01=6) legdad=3.
    if (ythpar01>=8) legdad=4.
else if (pinf096>0 or pinf159>0).

```

Appendix 3: Family Background Variable Creation

```
if (pinf096=2 or pinf159=2) legdad=1.  
if (pinf096=4 or pinf159=4) legdad=2.  
if (pinf096=6 or pinf159=6) legdad=3.  
if (pinf096>=8 or pinf159>=8) legdad=4.  
end if.  
  
*< use info from youth roster if paryouth info not  
available >  
do if any(paryth01, 0,2,4,6,8,10,12,14,16,18,20,22)  
and (pinf096=-4 and pinf159=-4) and (legdad=-4)  
and (ythpar01=-4) and (ythparid=-4).  
    if (paryth01=2) legdad=1.  
    if (paryth01=4) legdad=2.  
    if (paryth01=6) legdad=3.  
    if (paryth01>=8) legdad=4.  
end if.  
  
*****YOUTH'S CITIZENSHIP SECTION*****  
  
*determine citizenship of youth based on bioparent  
citizenship or youth's birth in the u.s.  
  
*<initialize everyone to can't determine ythcitiz=3.>  
compute ythcitiz=-4.  
  
do if (legmom=1 or legdad=1) and (ythcitiz=-4).  
    if (p2001=1) ythcitiz=1.  
    if (p2001 ne 1) and (p2008<=dob01y) ythcitiz=3.
```

```
        if (p2001 ne 1) and (p2008>dob01y) ythcitiz=2.  
end if.  
do if (splegmom=1 or splegdad=1) and (ythcitiz ne 1).  
    if (p2037=1) ythcitiz=1.  
    if (p2037 ne 1) and (p2044<=dob01y) ythcitiz=3.  
    if (p2037 ne 1) and (p2044>dob01y) ythcitiz=2.  
end if.  
do if (legmom ne 1 and legdad ne 1) and (ythcitiz ne 1).  
    if (p210101=1 or p210102=1) ythcitiz=1.  
    if (p210101 ne 1) and (p2108b01<=dob01y and  
        p2108b01>0) ythcitiz=3.  
    if (p210101 ne 1) and (p2108b01>dob01y)  
        ythcitiz=2.  
    if (p210101<0) and (p210102<0) and  
        (p2108b01<0) ythcitiz=3.  
end if.  
if (pinf015y=-4) ythcitiz=-4.  
  
*****hand edits using youth roster data when  
paryouth data missing*****  
if (pubid=1745 or pubid=5812 or pubid=6481 or  
    pubid=6482 or pubid=6483 or pubid=6748 or  
    pubid=6749) ythcitiz=3.  
if (pubid=4659 or pubid=4679 or pubid=7358)  
    ythcitiz=3.  
  
execute.
```

YOUTH'S RELATIONSHIP TO HOUSEHOLD PARENT FIGURE(S) AT AGE 2, 6, AND 12

Variables Created: CV_HH_REL_AGE_2
CV_HH_REL_AGE_6
CV_HH_REL_AGE_12

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
PUBID	PUBID	P818200-13	PC8-018.02_001-_014
dob01d, m, y	KEY!BDATE_D,_M,_Y	P818300-13	PC8-018.03_001-_014
P3051Y1-P3051Y6	P3-051.01_Y-.06_Y	P818400-13	PC8-018.04_001-_014
P305301-P305306	P3-053.01-.06	P818500-13	PC8-018.05_001-_014
P3067Y1-P3067Y6	P3-067.01_Y-.06_Y	P818600-13	PC8-018.06_001-_014
P3067Y2-5	P3-067.02_Y-.05_Y	P818700-13	PC8-018.07_001-_014
P818100-13	PC8-018.01_001-_014	P818800-13	PC8-018.08_001-_014
P8181000-13	PC8-018.10_001-_014	P818900-13	PC8-018.09_001-_014
P8181100-13	PC8-018.11_001-_014	PC801901-17	PC8-019.01-_17
P8181200-13	PC8-018.12_001-_014	PC802001-16	PC8-020.01-_16
P8181300-13	PC8-018.13_001-_014	PC802101-16	PC8-021.01-_16
P8181400-13	PC8-018.14_001-_014	PC802201-16	PC8-022.01-_16
P8181500-13	PC8-018.15_001-_014	PC803101-11	PC8-031.01-_11
P8181600-13	PC8-018.16_001-_014	PC803201-11	PC8-032.01-_11
P8181700-13	PC8-018.17_001-_014	PC803501-10	PC8-035.01-_10

Codes for Created Variable

1 = Both biological parents

2 = Biological mother, other parent present

3 = Biological father, other parent present

4 = Biological mother, marital status unknown

5 = Biological dad, marital status unknown

6 = Adoptive parent(s)

7 = Foster parent(s)

8 = Other adults, biological parent status unknown, not group quarters

9 = Group quarters

10 = Anything else

This program creates variables indicating the relationship of the youth to the primary adults in the household when the youth was age 2, 6, and 12.

```

array yrelate(3) yrelate2 yrelate6 yrelat12;
array continue(6) p305301-p305306;
array maryear(6) p3051y1-p3051y6;
array stopyear(6) p3067y1-p3067y6;
array stoplive(16) pc802201-pc802216;
array livewh(17, 14) p818100-p818113 p818200-p818213 p818300-p818313 p818400-p818413 p818500-
    p818513 p818600-p818613 p818700-p818713 p818800-p818813 p818900-p818913 p8181000-
    p8181013 p8181100-p8181113 p8181200-p8181213 p8181300-p8181313 p8181400-p8181413
    p8181500-p8181513 p8181600-p8181613 p8181700-p8181713;
array awayyear(10) pc803501-pc803510;
array awaywho(11) pc803201-pc803211;
array loopnum(3) loopnum2 loopnum6 loopnm12;
array respondt(17) pc801901-pc801917;
array noknow(17) pc802001-pc802017;
array refuse(17) pc802101-pc802117;
array place(11) pc803101-pc803111;
array lastnum(3) lastnum1 lastnum2 lastnum3;

/**three loops because of 2, 6, 12*/
do i= 1 to 3;

```

Appendix 3: Family Background Variable Creation

```
if i=1 then age=dob01y + 2;  
else if i=2 then age=dob01y +6;  
else if i=3 then age=dob01y +12;  
  
/** Initialize everyone to zero or -4 **/  
yrelate(i)=0;  
loopnum(i)=0;  
  
/* 1. The youth lived prior to live with parents */  
  
if age > 1900 then do;  
  
/**cases satisfied the following condition should have yrelate > 0. For age 2, the exception is those exiting from the  
first loop and stay with respondent or don't know or refuse this information; don't know or refuse in year first lived  
together; and don't know or refused the year relationship ended. If the year of the last loop larger than 0 but less  
than age then exit condition with the respondent then compute in the following sections. These cases still have  
yrelate=0 after this section and will be processed in section 2 or section 3. The same process is followed for age 6 and  
12.**/  
  
if pc8009y1 >=age or pc8009y1 in (-1,-2) then do;  
  
****a. identify the loop number into which age 2 (6,12) falls****/  
/**note that if there are multiple changes in the age year then the loopnum takes the last one**/  
/**there are 15 loops because of the 'j+1,' and the 17th loop is the respondent**/  
do j=1 to 15;  
  if stoplive(1) >=age then loopnum(i)=1;  
  if 0 < stoplive(j) < age and stoplive(j+1) >= age then loopnum(i)=j+1;  
/**the next line ensures that the last relationship reported during the age is the one used**/  
  if stoplive(j)=stoplive(j+1)=age then loopnum(i)=j+1;  
end;  
  
/**above deals with the 'good' data; below deals with the 'obnoxious' data**/  
/**max year of the prior loops**/  
maxyear=max(of pc802201-pc802216);  
  
/** taking care of the cases for pc8009y1 (year first lived)>=age and loopnum(i)=0 (exit from the first loop, age does  
not fall into any interval due to -1, -2 in stoplive or errors in reported date)**/  
if loopnum(i)=0 and pc8009y1 >=age then do;  
  /**note that if the loop ended in the first loop, then maxyear=-4, loopnum(i)=0;**/  
  /**the situation that maxyear=-4 and respondt(1)=1 (exit condition for 1st loop) will be taken care of later; takes  
care of the dk, refuse first**/  
  if maxyear=-4 and noknow(1)=1 then loopnum(i)=-2;  
  if maxyear=-4 and refuse(1)=1 then loopnum(i)=-1;  
/**reported error - dont know or refuse the year or give 'wrong' year (inconsistency)**/  
  if maxyear in (-1,-2) then loopnum(i)=maxyear;  
  else if 0 < maxyear < age then loopnum(i)=-1 ;  
  else if maxyear >=age then do k=1 to 16;  
    if stoplive(k) in (-1,-2) and loopnum(i) ^=-1 then loopnum(i)=stoplive(k);  
  end;  
end;  
  
/** taking care of the cases when pc8009y1 is -1 (-2) and loopnum=0**/  
if loopnum(i)=0 and pc8009y1 in (-1,-2) then do;  
  if maxyear in (-1,-2,-4) and (pc802601 >=age or pc802601 in (-1,-2,-4))  
    then loopnum(i)=pc8009y1;  /**for those with pc802601< age, loopnum=0(yrelate=0)**/  
  if maxyear >=age then loopnum(i)=pc8009y1;
```

```

counter1=0;
/**this determines the last loop reported by a respondent, regardless of whether the data are real, dont know or
refuse**/ 

if 0 < maxyear < age then do;
do j=1 to 16;    /**find out which loop is the last one ***/ 
if stoplive(j)=-4 and counter1=0 then do;
  lastloop=j-1;      counter1=counter1+1;
end;
end;
if j=16 and stoplive(16) ^=-4 then lastloop=16;

*****if stoplive(lastloop)>0 (valid year) and respondt(lastloop+1)=1  computed in one of the following sections -
2 or 3;
  if stoplive(lastloop)>0 and (noknow(lastloop+1)=1 or refuse(lastloop+1)=1) then loopnum(i)=pc8009y1;
  if stoplive(lastloop) in (-1,-2) then loopnum(i)=pc8009y1;
  end;
end;

/**b. recode the yrelate**/ 

if loopnum(i) in (-1,-2) then yrelate(i)=loopnum(i);

if loopnum(i) > 0 then do;
/** lived with bio. mom, marital status unknown ***/ 
if livewh(loopnum(i),2 )=1 then yrelate(i)=4;

/** lived with bio dad, marital status unknown**/ 
if livewh(loopnum(i),3 )=1 then yrelate(i)=5;

/** lived with both biological parents**/ 
if livewh(loopnum(i),2 )=1 and livewh(loopnum(i),3)=1 then yrelate(i)=1;

/** lived with adoptive parents**/ 

/** lived with other adults, bio. parent status unknown, not group qtrs ***/ 
if livewh(loopnum(i),4 )=1 then yrelate(i)=8;
if livewh(loopnum(i),5 )=1 then yrelate(i)=8;

/** lived with foster parents**/ 
if livewh(loopnum(i),7 )=1 then yrelate(i)=7;

/**lived in group quarters**/ 
do k = 8,9,10,11,12;
  if livewh(loopnum(i),k )=1 then yrelate(i)=9;
end;

/** any other living arrangement**/ 
do k = 6,13,14;
  if livewh(loopnum(i),k )=1 then yrelate(i)=10;
end;
end;
end;

/* 2. check for periods spent away from parents, pc802601 <age. Note that if the youth lived with respondent at
age 2, 6, 12, then yrelate=0. These cases will be coded in section 3. */

```

Appendix 3: Family Background Variable Creation

```
if (pc802501=1) and (pc802601 < age ) and yrelate(i)=0 then do;  
/****note that if pc802601=-1 or -2, then the case is included****/  
  
/***find out which loop is the last one; there is no change after that; think of counter 2 as a flag ***/  
counter2=0;  
do j=1 to 10;  
if awayyear(j)=-4 and counter2=0 then do;  
    lastchg=j-1;  
    if j=1 then do;  
        lastchg=1;  
        nochange=1;  
        end;  
    counter2=counter2+1;  
end;  
end;  
if awayyear(10)>-4 then lastchg=10;  
  
/***find out into which loop the age falls ***/  
do j=1 to lastchg;      /***10 equals to the maximum complete spells away from parents***/  
if j=1 and awayyear(1)=-4 then lastnum(i)=1;  
if j=1 and awayyear(1) >= age then lastnum(i)=1;  
if j < lastchg and 0< awayyear(j) < age and awayyear(j+1) >= age then lastnum(i)=j+1;  
if j < lastchg and awayyear(j)=awayyear(j+1)=age then lastnum(i)=j+1;  
if j=lastchg and 0 < awayyear(lastchg)< age then lastnum(i)=lastchg+1;  
end;  
  
if lastnum(i)=-1 then lastnum(i)=-1; /*if there is -1/-2 in awayyear, no such cases in current data**/  
  
/***r goes on to give more data, but the year info in the first loop is invalid***/  
if lastnum(i)=1 and nochange ^= 1 and pc802601 in (-1,-2) then yrelate(i)=pc802601;  
/***yrelate=-1 or -2**/  
if lastnum(i)=-1 then yrelate(i)=-1;  
if yrelate(i) =0 then do;  
    if place(lastnum(i)) in (-1,-2) then yrelate(i)=place(lastnum(i));  
    if place(lastnum(i)) in (1,4) then yrelate(i)=10; /***coded as 'anything else'**/  
    if place(lastnum(i))=3 then yrelate(i)=9;      /***group quarters**/  
    if place(lastnum(i))=2 then do;  
/**if respondent (1) is coded [awaywho(lastnum(i))=1] then go to section 3 to determine marital status**/  
        if awaywho(lastnum(i)) = 2 then yrelate(i)=1;  
        if awaywho(lastnum(i)) = 3 then yrelate(i)=4;  
        if awaywho(lastnum(i)) = 4 then yrelate(i)=5;  
        if awaywho(lastnum(i)) in (5,6,11) then yrelate(i)=8;  
        if awaywho(lastnum(i)) in (7,8,9) then yrelate(i)=10;  
        if awaywho(lastnum(i)) =10 then yrelate(i)=7;  
        if awaywho(lastnum(i))in (-1,-2) then yrelate(i)=awaywho(lastnum(i));  
    end;  
end;  
end;  
  
/* 3. The youth lived with parents continuously or the first separation is >= age or the separation time is with  
respondent. Flag2 = 1 indicates that a marriage happened before or when youth is 2 (6,12) and ended when or after  
youth is 2 */  
  
if yrelate(i)=0 and pc802501 in (-1,-2) then yrelate(i)=pc802501;  
if yrelate(i)=0 and ( (pc802501=0) or (pc802501=1 and pc802601 >= age) or
```

Appendix 3: Family Background Variable Creation

```
( (pc802501=1) and (pc802601 < age) ) then do;
do j = 1 to 6;*** Determine if the youth has lived with both parents since living with the respondent ***
  if continue(j)=1 and ( 0 <= dob01y - maryear(j) < 1900 ) and ythpar01 in (1,2) and yth2bios=1 then do;
    yrelate(i)=1;
  end;
end;

if yrelate(i)=0 then do;
  if max(of p3051y1-p3051y6)=-4 then flag1=-4;
  do j=1 to 6;
    if j = < 5 and (1900 < maryear(j) < age) and (stopyear(j) >=age) then flag1=1;
    else if j=6 and (1900 < maryear(j) < age) then flag1=1;
  end;
  if flag1=. then flag1=-1; /*for those with -1,-2 in the age period or the age is not with any marriages */
  if flag1=1 then do;
    if ythpar01=1      then yrelate(i)=2;           *** bio. mom, other parent figure present ***
    else if ythpar01=2  then yrelate(i)=3;           *** bio. dad, other parent figure present ***
    end;
  else if flag1 in (-1,-4) then do;
    if ythpar01=1 then yrelate(i)=4;
    else if ythpar01=2 then yrelate(i)=5;
    end;
  if ythpar01 in (3,4) then yrelate(i)=6;           *** adoptive parents ***
  else if ythpar01 in (5,7,11,15,17,19,21,6,8,12,16,18,20,22,99) then yrelate(i)=8;
    *** other adults, bio. parent status unknown ***
  else if ythpar01 in (9,13,10,14) then yrelate(i)=7; *** foster/temporary parents ***
  end;
end;

end;
end;

if pc800401=-4 then yrelate2=-4;
if pc800401=-4 then yrelate6=-4;
if pc800401=-4 then yrelat12=-4;

*hand edits;
if pubid=228 then yrelate2=5 and yrelate6=5 and yrelat12=5;
if pubid=1075 then yrelate2=8 and yrelate6=8 and yrelat12=8;
if pubid=1128 then yrelate2=4 and yrelate6=4 and yrelat12=4;
if pubid=3404 then yrelate2=1 and yrelate6=1 and yrelat12=1;
if pubid=8020 then yrelate2=5 and yrelate6=5 and yrelat12=5;

endsas;
```

